

Europäisches Patentamt

European Patent Office

Office européen des brevets



(11)

EP 1 014 730 A1

(12)

EUROPEAN PATENT APPLICATION

(43) Date of publication:
28.06.2000 Bulletin 2000/26

(51) Int Cl.7: H04N 7/66, H04N 7/24,
H04N 7/52

(21) Application number: 99310433.0

(22) Date of filing: 22.12.1999

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(72) Inventors:
• Wan, Xiang
Waterloo Ontario, N2L 3B8 (CA)
• Morin, Marc H.
Ontario, N2T 2A3 (CA)

(30) Priority: 23.12.1998 US 219424

(74) Representative: Gibson, Stewart Harry
URQUHART-DYKES & LORD,
Three Trinity Court,
21-27 Newport Road
Cardiff CF24 0AA (GB)

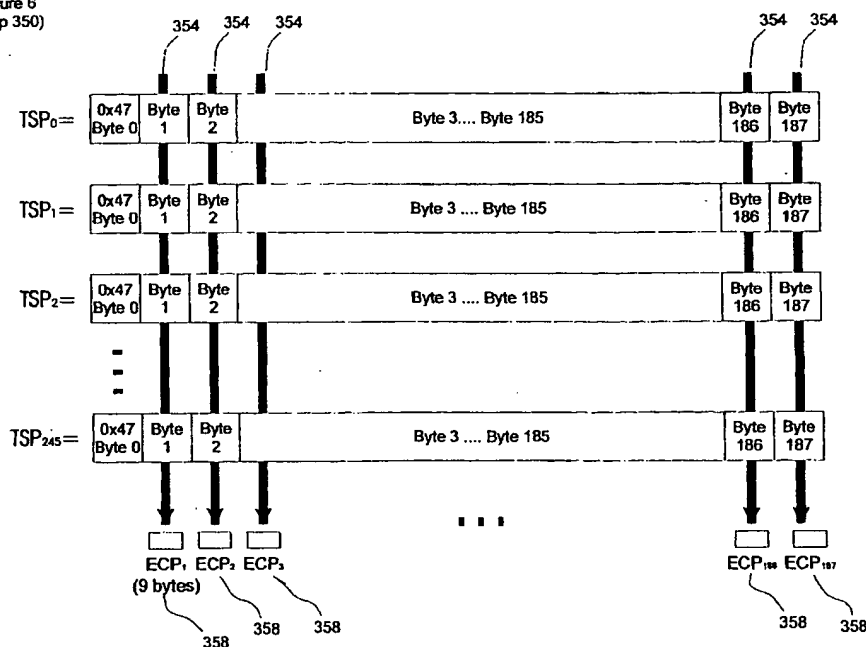
(71) Applicant: Pixstream Incorporated
Waterloo, Ontario N2L 3L3 (CA)

(54) Forward error correction at MPEG-2 transport stream layer

(57) The invention is for a system and method for performing forward error correction on an MPEG-2 transport stream that has optional error correction information embedded within the transport stream. An error correction operation is performed on a super group of packets within the transport stream. The error correction data is inserted into a created trailer group of MPEG-2 compliant transport stream packets having no payload,

but only an adaptation field. The trailer group packets are optional and therefore can be discarded by a standard MPEG-2 decoder, but can also be used to recover data lost in the transmission of the transport stream. The invention results in a system and method to correct an MPEG-2 transport stream that can be used in any one of the digital video broadcast (DVB) formats, without the need for error correction decoders which are specific to the DVB format.

Figure 6
(Step 350)



Description**Field of the Invention**

- 5 **[0001]** This invention relates to techniques of forward error correction with a communications protocol, and in particular, relates to forward error correction of MPEG-2 at the transport stream layer.

Background of the Invention

- 10 **[0002]** Digital video includes the steps of the encoding of moving pictures into a digital signal, broadcasting the digital signal, and decoding the digital signal into a viewable format such as a television screen or a computer monitor. With the rapid improvements in computer technology, digital video is now readily available for computer systems, etc. and will soon be widely available for broadcasting. As digital video can also incorporate the transmission of audio signals, it is believed that this technology will eventually replace current television technology (analog video), and will have
15 broad application in many areas such as the Internet and telecommunications in general.

- [0003]** Because of the broad application of digital video, the International Organization For Standardization (ISO) formed ISO/IEC JTC1 SC29 WG11, more commonly known as the Motion Pictures Expert Group 2 (MPEG-2) to generate standards for broadcast quality digital video. (An earlier committee was ISO CD 11172, or MPEG-1, which created the first generation of digital video standards, which are optimised for CD-ROM. Another variant under development
20 is MPEG-4 for low bandwidth video telephony.) In particular, MPEG-2 defines a compressed bit stream, but the actual implementation of the compression operations are up to individual manufacturers of digital video equipment.

[0004] MPEG-2 is a standard ("MPEG-2 standard") with various parts at different levels of completion. Of particular interest are:

- 25 ISO/IEC DIS 13818-1 Information technology --

Generic coding of moving pictures and associated audio information:
Systems

- 30 ISO/IEC DIS 13818-2 Information technology --

Generic coding of moving pictures and associated audio information:
Video

- 35 ISO/IEC 13818-3:1995 Information technology --

Generic coding of moving pictures and associated audio information --
Part 3: Audio

- 40 and the contents of each of these references are incorporated herein by reference. The MPEG-2 standard addresses the combining of one or more elementary streams of video, audio and other data into single or multiple streams which are suitable for storage or transmission. In very general terms, the MPEG-2 standard for transmitting digital video and associated audio and other information involves the following steps: first, a digital video signal (from a digital camera or from an analog to digital converter) is compressed by analyzing and encoding the signal using spatial and temporal
45 redundancy. Spatial redundancy refers to the redundant information inside one video frame while temporal redundancy refers to the redundant information between consecutive frames. This process generates: Intra-frames (I-frames), which contain all of the information in an entire image; Predicted frames (P-Frames), which have some compression as they are predicted based on past I-frames and/or other P-frames; and Bi-directional predicted frames (B-frames), which are the most compressed images as they are predicted from past and future I-Frames and P-Frames. Next, an
50 audio signal is compressed by removing low power tones adjacent high power tones. Removal of these tones does not affect the signal, because the high power tones tend to mask the lower power tones, making them inaudible to the human ear. Finally, the compressed video signals, audio signals and related time stamps of those signals are assembled into packets and inserted into a Packetized Elementary Stream (PES). Each packet in a packetized elementary stream contains overhead information such as a start code, stream id, packet length, optional packetized elementary stream
55 header and stuffing bytes, in addition to the actual packet bytes of video and audio data.

[0005] It will be apparent from the foregoing that the packetized elementary stream contains encoded video and audio data for storage or transmission. In order to multiplex several streams of packetized elementary streams, a Programme Specific Information (PSI) table is also created, which includes a series of tables to reassemble specific

packetized elementary stream within multiple channels of packetized elementary streams. Thus, the packetized elementary stream and program specific information form the basis for constructing either a Program Stream (PS) or a Transport Stream (TS) of packetized elementary stream and program specific information packets, which are particularly defined in ISO/IEC International Standard 13818-1.

[0006] The Program Stream is generally suitable for communicating or storing one program of coded data and other data in error free environments, such as an optical storage device, and where processing of system coding is a major consideration. In contrast, the Transport Stream is suitable for storing or communicating one or more programs in environments where significant errors can occur, such as packet-switched networks, etc. These errors can be manifested as bit value errors within a packet, or the loss of one or more entire packets.

[0007] However, while the transport stream was intended to protect data within the transport stream from corruption in error-prone environments, the transport stream has been found to be less than satisfactory in this regard and it remains difficult to ensure reliable transmission of the transport stream through these environments. These difficulties are more particularly explained in "Protecting MPEG-2 : FEC schemes in DVB-C,S,T", ("Protecting MPEG-2") © Copyright 1994-1998 Hewlett-Packard Company, the contents of which are incorporated herein by reference, and a copy of which is available at:

[0008] http://www.tmo.hp.com/tmo/pia/component_tes/PIAApp/Notes/English/MPEGpaper2.html. An example of error prone environments in which the transport stream can be broadcast are the various Digital Video Broadcasting Project (DVB) formats, such as Direct Broadcast Satellite (DBS), terrestrial cable networks (including coax and fibre optic etc.) or terrestrial broadcasts. In order to protect the transport stream and to achieve a Bit Error Ratio (BER) of no more than 10^{-11} , Forward Error Correction (FEC) is now often added to protect the transport stream during transmission. In simple terms, the method of correcting in these formats is as follows: the original transport stream is inserted into a forward error correction encoder; the resulting signal is broadcast over the environment to a receiver; and the transport stream is recovered from the received signal by a symmetric forward error correction decoder.

[0009] Generally, a different forward error correction encoder and decoder is required for each DVB format, and this requires that each receiver have a symmetric forward error correction decoder. Variations between forward error correction encoding techniques, such as Reed-Solomon (RS), include the actual amount of error correction code added to the end each transport stream packet, whether the forward error correction packets are interleaved or spliced, and whether viterbi convolutional coding is performed. In addition to the necessity of a symmetric forward error correction decoder, it will be apparent that the addition of error correction code to the end of each packet effectively adds another layer to the protocol stack and this new layer is specific to the transmission architecture and not subject to the MPEG-2 standard. Thus, a broadcaster must ensure that each intended receiver has a symmetric forward error correction decoder.

Summary of the Invention

[0010] It is an object of the present invention to provide a novel forward error correction method and system to protect a digital transport stream which obviates or mitigates at least one of the disadvantages of the prior art.

[0011] In a first aspect of the present invention, there is provided a method of creating a portion of an enhanced MPEG compatible data stream with optional forward error correction information included therein, the method comprising the steps of:

receiving an MPEG data stream of data packets from an MPEG data stream source;
forming a super group including a predefined number of the data packets;
performing a selected forward error correcting operation on the super group of packets to obtain error correction data;
creating index data which relates the error correction data to one or more corresponding super group data packets;
creating a trailer group including a sufficient number of MPEG compliant packets to accommodate the error correction data and the index data, the trailer group further including code to identify each packet as optional;
arranging the error correction data and the index data into the trailer group; and
assembling the MPEG compatible data stream with at least one super group followed by at least one trailer group corresponding with the at least one super group.

[0012] In particular, the assembly according to the method of the first aspect comprises two super groups followed by two corresponding trailer groups.

[0013] Alternately, the assembly according to the method of the first aspect comprises the super group followed by a predefined number of null packets and is then followed the trailer group. Typically, the predefined number is equivalent to an estimated maximum burst error of the broadcast environment.

[0014] Typically, the data stream according to the first aspect is an MPEG-2 transport stream; the super group com-

prises two-hundred-and-forty-six data packets; the trailer group comprises eleven data packets; and the forward error correction operation is Reed Solomon coding. It is generally preferred that the error correction operation be performed on parallel bytes of each packet within the super group. As found in the MPEG-2 standard, it is preferable that each of the packets in the super group includes a packet identifier and each of the packets in the trailer group includes a packet identifier different from the super group packet identifiers.

[0015] In a second aspect of the invention, there is provided a method of creating a portion of an enhanced MPEG compatible data stream with optional forward error correction information included therein, comprising the steps of:

receiving an MPEG data stream of data packets from an MPEG data stream source;
 creating at least one MPEG compatible packet with no data included therein;
 creating an unused portion of each packet received by shifting a portion of payload data in each of the received packets into the at least one created MPEG compatible packet, the created at least one MPEG compatible packets also having an unused portion;
 performing a predefined forward error correcting operation on the payload data of each packet from the previous step to obtain forward error correction data; and
 arranging the error correction data in the unused portions so as to interleave the error correction data with the payload data.

[0016] Typically, the data stream in the second aspect is an MPEG-2 transport stream. It is also generally preferred that the forward error correction operation is Reed Solomon coding.

[0017] In a third aspect of the present invention, there is provided a method of creating a portion of an enhanced MPEG compatible data stream with optional forward error correction information included therein, comprising the steps of:

receiving an MPEG data stream of data packets from an MPEG data stream source;
 forming a super group comprising a predefined number of packets;
 forming a plurality of sub-groups comprising a predefined number of packets from the super group;
 creating at least one MPEG compatible packet with no data included therein;
 creating an unused portion in a first packet of each sub-group by shifting a portion of payload data in each of the first packets into the at least one created MPEG compatible packet created in the previous step, the created packets being reinserted into the MPEG data stream being received in the first step;
 embedding an identifier within the unused portion of the first packet of each sub-group;
 performing a selected forward error correcting operation on the super group of packets to obtain error correction data;
 creating index data which relates the error correction data to one or more corresponding sub-group data packets in relation to the embedded identifiers;
 creating a trailer group including a sufficient number of MPEG compliant packets to accommodate the error correction data and the index data, the trailer group further including code to identify each packet as optional;
 arranging the error correction data and the index data into the trailer group; and
 assembling the MPEG compatible data stream with at least one super group followed by at least one trailer group corresponding with the at least one super group.

[0018] Typically, the data stream in the third aspect is an MPEG-2 transport stream. Further, it is generally preferred that the super group comprises two-hundred-and-forty-six data packets, each sub-group comprises six data packets, the trailer group comprises ten data packets, and the forward error correction operation is Reed Solomon coding performed on parallel bytes of each packet within the super group.

[0019] It is also typical in the method of the third aspect that each of the packets in the super group include a packet identifier and each of the packets in the trailer group include a packet identifier different from the super group packet identifiers.

[0020] In a fourth aspect of the present invention, there is provided a method of creating a portion of an enhanced MPEG compatible data stream with optional forward error correction information included therein, comprising the steps of:

receiving an MPEG data stream of data packets from an MPEG data stream source;
 forming a super group comprising a predefined number of packets;
 creating at least one MPEG compatible packet with no data included therein;
 creating an unused portion in each super group packet by shifting a portion of payload data in each of the super group packets into the at least one created MPEG compatible packet, the created packets being reinserted into

the MPEG data stream being received in the first step;
 embedding an identifier within the unused portion of each of the super group packets;
 performing a selected forward error correcting operation on the super group of packets to obtain error correction data;

5 creating index data which relates the error correction data to one or more corresponding super group data packets in relation to the embedded identifiers;

creating a trailer group including a sufficient number of MPEG compliant packets to accommodate the error correction data and the index data, the trailer group further including code to identify each packet as optional;

arranging the error correction data and the index data into the trailer group; and,

10 assembling the MPEG compatible data stream with at least one super group followed by at least one trailer group corresponding with the at least one super group.

[0021] In particular, the assembly according to the method of the fourth aspect comprises two super groups followed by two corresponding trailer groups.

15 **[0022]** Alternately, the assembly according to the method of the first aspect comprises the super group followed by a predefined number of null packets and is then followed the trailer group. Typically, the predefined number is equivalent to an estimated maximum burst error of the broadcast environment.

[0023] Typically, the data stream in the fourth aspect is an MPEG-2 transport stream, the super group comprises two-hundred-and-forty-six data packets, the trailer group comprises eleven data packets, the forward error correction operation is Reed Solomon coding and the error correction operation is performed on parallel bytes of each packet within the super group.

[0024] It is also typical in the method of the fourth aspect that each of the packets in the super group include a packet identifier and each of the packets in the trailer group include a packet identifier different from the super group packet identifiers.

25 **[0025]** In a fifth aspect of the present invention, there is provided a method of transmitting data as an enhanced MPEG compatible data stream with optional forward error correction information included therein, the method comprising the steps of:

receiving an MPEG data stream of data packets from an MPEG data stream source;

30 forming a super group including a predefined number of the data packets;

performing a selected forward error correcting operation on the super group of packets to obtain error correction data;

creating index data which relates the error correction data to one or more corresponding super group data packets;

35 creating a trailer group including a sufficient number of MPEG compliant packets to accommodate the error correction data and the index data, the trailer group further including code to identify each packet as optional;

arranging the error correction data and the index data into the trailer group;

assembling the enhanced MPEG compatible data stream with at least one super group followed by at least one trailer group corresponding with the at least one super group;

broadcasting the enhanced MPEG compatible data stream;

40 receiving the enhanced MPEG compatible data stream;

extracting at least one received super group and at least one received trailer group corresponding to the at least one received super group from the data stream received in the previous step; and

performing an error recovery operation on the at least one received super group using the error correction data in the at least one corresponding received trailer group.

45 **[0026]** Typically, the data stream of the fifth aspect is an MPEG-2 transport stream, the forward error correction operation is Reed Solomon coding, and the error correction operation is performed on parallel bytes of each packet within the super group.

[0027] It is also typical in the method of the fifth aspect that each of the packets in the super group include a packet identifier and each of the packets in the trailer group include a packet identifier different from the super group packet identifiers.

50 **[0028]** In a sixth aspect of the invention, there is provided a method of recovering a portion of an enhanced MPEG compatible data stream of data packets having super groups of data packets and corresponding trailer groups of optional packets having forward error correction information and indexing information included therein, comprising the steps of:

receiving the enhanced MPEG compatible data stream;

extracting at least one received super group and at least one received trailer group corresponding to the at least

one received super group from the received data stream;
 performing an error recovery operation on the at least one received super group using the error correction data in
 the at least one corresponding received trailer group;

5 [0029] Typically, the data stream of the sixth aspect is an MPEG-2 transport stream, the super group comprises two-hundred-and-forty-six data packets, the trailer group comprises eleven data packets, and the error recovery operation is Reed Solomon decoding.

[0030] It is also typical in the method of the sixth aspect that each of the packets in the super group include a packet identifier and each of the packets in the trailer group include a packet identifier different from the super group packet
 10 identifiers.

[0031] In a seventh aspect of the invention, there is provided a system for creating a portion of an enhanced MPEG compatible data stream with optional forward error correction information included therein, comprising:

15 a receiver for receiving an MPEG data stream of data packets from an MPEG data stream source;
 processing means for forming a super group including a predefined number of the data packets;
 processing means for performing a selected forward error correcting operation on the super group of packets to obtain error correction data;
 processing means for creating index data which relates the error correction data to one or more corresponding super group data packets;
 20 processing means for creating a trailer group including a sufficient number of MPEG compliant packets to accommodate the error correction data and the index data, the trailer group further including code to identify each packet as optional;
 processing means for arranging the error correction data and the index data into the trailer group;
 processing means for assembling the enhanced MPEG compatible data stream with at least one super group
 25 followed by at least one trailer group corresponding with the at least one super group; and
 a transmitter for transmitting the enhanced MPEG compatible data stream.

[0032] In a particular system according to the seventh aspect, the MPEG source is integral with the receiver.

[0033] In an eighth aspect of the present invention, there is provided a system for recovering a portion of an enhanced
 30 MPEG compatible data stream of data packets having super groups of data packets and associated trailer groups of optional packets having forward error correction information and indexing information included therein, comprising:

a receiver for receiving the enhanced MPEG compatible data stream;
 a processing means for extracting at least one received super group and at least one received trailer group corresponding to the at least one received super group from the received data stream;
 35 a processing means for performing an error recovery operation on the at least one received super group using the error correction data in the at least one corresponding received trailer group; and
 a transmitter for transmitting said recovered super group to an MPEG end-user.

40 [0034] In a particular system according to the seventh aspect, the MPEG end-user is integral with the transmitter.

[0035] In a ninth aspect of the invention, there is provided a portion of an enhanced MPEG compatible data stream with optional forward error correction information included therein, comprising:

45 at least one super group including a predefined number of data packets;
 at least one trailer group including a predefined number of MPEG compliant packets, the at least one trailer group further including code to identify each packet as optional;
 error correction data embedded within the at least one trailer group, the error correction data being obtained from a selected forward error correcting operation performed on at least one corresponding super group of data packets;
 and
 50 index data embedded within the at least one trailer group which relates said error correction data to one or more said corresponding super group data packets.

[0036] In particular, the data stream according to the ninth aspect is typically an MPEG-2 transport stream. Further, the super group in the ninth aspect typically comprises two-hundred-and-forty-six data packets, the trailer group comprises eleven data packets, the forward error correction operation is Reed Solomon coding, and the error correction data was generated from parallel bytes of each packet within said super group.
 55

[0037] It is also typical in the method of the ninth aspect that each of the packets in the super group include a packet identifier and each of the packets in the trailer group include a packet identifier different from the super group packet

identifiers.

[0038] The present invention provides a novel method and system for taking an MPEG data stream generated by an MPEG encoder and creating an enhanced MPEG compliant data stream having optional error correction data included therein, such that an asymmetric decoder or end-user can process the data contained within the original MPEG data stream and will ignore the error correction data and associated index data contained in the enhanced transport stream. However, a symmetric decoder can use the error correction to correct for errors in the received enhanced transport stream.

Brief Description of the Drawings

[0039] The present invention will now be described, by way of example only, with reference to certain embodiments shown in the attached Figures in which:

Figure 1 shows a schematic representation of the syntax of an MPEG-2 Transport Stream (TS) Packet;
 Figure 1a shows a schematic representation of the syntax of an adaptation field in accordance with the Transport Stream Packet of Figure 1;
 Figure 1 b shows a schematic representation of payload data in accordance with the Transport Stream Packet of Figure 1;
 Figure 2 shows a schematic representation of a prior art system for forward error correcting an MPEG-2 Transport Stream (TS);
 Figure 3 shows a schematic representation of a system for forward error correcting an MPEG-2 Transport Stream (TS) in accordance with an embodiment of the present invention;
 Figure 4 shows a flowchart of a method of forward error correcting an MPEG-2 Transport Stream (TS) in accordance with an embodiment of Figure 3;
 Figure 5 shows a schematic representation of defining a super group of transport stream packets in accordance with the method of Figure 3;
 Figure 6 shows a schematic representation of performing an error correction operation on parallel bytes within each packet of the super group in accordance with the method of Figure 4;
 Figure 7 shows a schematic representation of creating a trailer group at the end of the super group shown in Figure 5 and in accordance with the method of Figure 4;
 Figure 8 shows a schematic representation of defining the trailer group of Figure 7 to be optional and to have no payload data;
 Figure 9 shows a schematic representation of embedding the error correction information shown in Figure 8 into the first nine transport stream packets of the trailer group shown in Figure 7, and in accordance with the method of Figure 4;
 Figure 10 shows a schematic representation of embedding error correction information into the second last packet of the trailer group shown in Figure 7, and in accordance with the method of Figure 4;
 Figure 11 shows a schematic representation of storing index information into the second last packet of the trailer group shown in Figure 7, and in accordance with the method of Figure 4;
 Figure 11a shows a schematic representation of storing sequence indices into the second last packet of the trailer group shown in Figure 7, and in accordance with the method of Figure 4;
 Figure 11b shows a schematic representation of storing sequence indices into the last packet of the trailer group shown in Figure 7, and in accordance with the method of Figure 4;
 Figure 12a shows a first alternative of transmitting an enhanced transport stream comprising a first super group, a second super group, a first trailer group and a second trailer group;
 Figure 12b shows a second alternative of transmitting an enhanced transport stream comprising a super group, null packets and a trailer group; and,
 Figure 13 shows a flowchart of a method of error recovery from an enhanced transport stream symmetric to the method of Figure 4.

Detailed Description of the Drawings:

[0040] Before discussing the present invention, an MPEG-2 Transport Stream (TS) packet will be discussed briefly. An MPEG-2 transport stream packet (TSP) in accordance with ISO Standard 13818-1 is indicated generally at 30 in Figure 1. Transport stream packet 30 is fixed in length at one-hundred-and-eighty-eight bytes and comprises a header section 34 and an information section 38. Header section 34 is fixed at four bytes in length and comprises a sync byte 42, a transport error indicator bit 46, a payload start indicator bit 50, a transport priority bit 54, a thirteen-bit Packet Identifier (PID) 58, two scrambling control bits 62, two adaptation field control bits 66, and four continuity counter (CC)

bits 70.

[0041] Packet Identifier 58 indicates the type of data stored in the packet payload, such as packetized elementary stream or program specific information. Adaptation field control bits 66 indicate what type of information section 38 follows header 34. Continuity Counter 70 is incremented for each transport stream packet 30 with the same Packet Identifier 48 and wraps around to zero after reaching its maximum value. According to ISO Standard 13818-1, Continuity Counter 70 does not increment when the adaptation field control 66 equals "b00" or "b10" (As used herein, "b..." denotes a number expressed in binary notation). The remaining fields in header section 34 are explained in detail in ISO Standard 13818-1.

[0042] Information section 38 is fixed at one-hundred-and-eighty-four bytes in length, and comprises an adaptation field 74 and/or payload data 78. For greater clarity, information section 38 contains either: 1) an adaptation field 74 having a length of one-hundred-and-eighty-four bytes; 2) payload data 78 having a length of one-hundred-and-eighty-four bytes; or 3) an adaptation field 74 followed by payload data 78, each having variable lengths, and having a combined length of one-hundred-and-eighty-four bytes.

[0043] Referring now to Figure 1a, adaptation field 74 comprises an adaptation field length 82, flags 84, and data fields 90. Adaptation field length 82 specifies the number of bytes in adaptation field 74 immediately following adaptation field length 82. Flags 84 comprise eight, one-bit flags including a discontinuity indicator flag 86, a random access indicator flag 88, an elementary stream priority indicator flag 94, a Program Clock Reference (PCR) flag 98, an Original Program Clock Reference Base (OPCR) flag 102, a Splicing Point flag 106, a Transport Private Data flag 110, and an Adaptation Field Extension flag 114. Data fields 90 have a maximum length of one-hundred-and-eighty-two bytes and contain several fields corresponding to their respective flags 84, which are defined only if their respective flag 84 is set to a value of one. Data fields 90 comprise a PCR field (forty-two bits) 118, an OPCR field (forty-two bits) 122, a Splice Countdown field (eight bits) 126, a one-byte Private Data Length field 128, a Private Data Field (Variable between zero to one-hundred-and-eighty-one bytes in length) 132, a one-byte adaptation extension length field 136, three one-bit flags 140, optional fields 152 corresponding with flags 140 and one or more stuffing bytes 156. Each of the fields in adaptation field 74 are explained in detail in ISO Standard 13818-1.

[0044] Referring now to Figure 1b, payload 78 comprises one or more data bytes 160, up to a maximum length of one-hundred-and-eighty-four bytes depending on the length of adaptation field 74. Data bytes 160 represent a series of contiguous data bytes of: 1) Packetized Elementary Stream (PES) packets, 2) Program Specific Information (PSI) Sections; 3) Packet Stuffing Bytes following PSI Sections; or 4) Private Data not in these structures as indicated by Packet Identifier 58.

[0045] Having described transport stream packet 30, a prior art forward error correction system will now be discussed, and is indicated generally at 200 in Figure 2. Forward error correction system 200 comprises a forward error correction encoder 204 for receiving an MPEG-2 transport stream 208 from a source such as an MPEG-2 encoder, and for outputting an error corrected data stream (ECDS) 212 to a transmitter 216 such as a radio tower, satellite dish, cable head-end or any DVB transmitter as is known in the art. Transmitter 216 is connected via a broadcast environment 218 to a complementary receiver 220, which in turn is connected to a decoder 224. Decoder 224 is symmetric to encoder 204, and outputs transport stream 208 to MPEG-2 end-user 228. MPEG-2 end-user 228 can be a personal computer, a digital television, a storage device or any higher layer medium that can decode or otherwise use transport stream 208.

[0046] The operation of prior art system 200 will now be discussed with reference to Figure 2. An MPEG-2 transport stream source, such as an MPEG-2 encoder, creates transport stream 208, which is received by forward error correction encoder 204. Forward error correction encoder 204 interleaves each transport stream packet and performs a forward error correcting operation on each transport stream packet. Any error encoding operation known in the art can be used, but the Reed-Solomon operation is generally the most common. Next, the error correcting code is added to the end of each transport stream packet, thus creating a series of error corrected packets ($ECP_0 \dots ECP_N$). Together, these packets essentially form a new protocol layer indicated as an error corrected data stream (ECDS) 212.

[0047] Error corrected data stream 212 is then inputted into transmitter 216, which broadcasts error corrected data stream 212 via the broadcast environment to receiver 220. During broadcast, error corrected data stream 212 can lose information due to errors caused by unreliability of environment 218. Error corrected data stream 212 is then inputted into decoder 224 which extracts transport stream 208 from error corrected data stream 212, and uses the error correcting code appended to the end of each transport stream packet, to correct for errors in ECP_0 through ECP_N . The extracted transport stream 208 is then reordered from its interleaved state and back into its correct order, and outputted to MPEG-2 end-user 228.

[0048] It will now be apparent that transport stream packet 30 does not contemplate error correction coding, and thus prior art systems of protecting transport stream effectively add another protocol layer, thereby requiring a decoder 224 symmetric to each encoder 204 and specific to the broadcast medium used. As a result, systems without decoder 224 cannot use or otherwise decode error corrected data stream 212 at all.

[0049] An embodiment of the present invention will now be discussed with reference to Figures 3 to 13. A system

of forward error correcting an MPEG-2 Transport Stream (TS) in accordance with a presently preferred embodiment of the present invention is indicated generally at 250 in Figure 3. Forward error correction system 250 comprises an enhanced forward error correction encoder 254 for receiving an MPEG-2 Transport Stream 208 from any suitable source such as an MPEG-2 encoder, and for outputting an enhanced transport stream (TS⁺) 258 to a transmitter 216 such as a radio tower, satellite dish, cable head-end or any DVB transmitter as is known in the art. Transmitter 216 is connected via a broadcast environment 218 to a complementary receiver 220, which in turn can be connected to a decoder 262. Decoder 262 is symmetric to encoder 204, and outputs transport stream 208, to MPEG-2 end-user 228. However, because enhanced transport stream 258 is MPEG-2 compliant, receiver 220 can output enhanced transport stream 258 directly to MPEG-2 end-user 228, bypassing decoder 262.

[0050] The operation of system 250 will now be discussed with reference to Figure 3. Transport stream 208 is received by forward error correction encoder 254. As will be discussed in greater detail below, encoder 254 performs a forward error correcting operation on parallel portions of a predefined super group of packets. Next, a predefined trailer group comprising a number of optional MPEG-2 compliant transport stream packets are created, and the error correcting code is embedded within the private data fields 132 of the new packets. The end trailer group packets contain identifiers or addressing information to correlate the error correcting code with the appropriate packet in the super group. As will be explained in greater detail below, it is presently preferred that two sequential super groups should be followed by their two corresponding trailer groups in the formation of a portion of enhanced transport stream 258, in order to protect for bursty errors. In another alternative, each trailer group packet can instead be simply appended to the end of its corresponding super group, to form a portion of enhanced transport stream 258. In this latter alternative, it is contemplated that a series of optional null transport stream packets containing stuffing bytes can be added between the super group and its trailer group in order to increase protection for bursty errors. The process for either alternative is repeated to form a continuous enhanced transport stream 258, which is then outputted to transmitter 216.

[0051] Enhanced transport stream 258 is then broadcasted by transmitter 216 via a broadcast environment 218 to receiver 220. During broadcast, the unreliability of the environment can cause portions of transport stream packets or entire transport stream packets within enhanced transport stream 258 to be lost. However, because enhanced transport stream 258 is MPEG-2 compliant, it can be passed directly to MPEG-2 end-user 228, which will ignore the optional trailer group and process the remaining transport stream packets according to MPEG specifications. However, enhanced transport stream 258 can also be inputted into decoder 262, if present, which recovers transport stream 208 from enhanced transport stream 258. As will be discussed in greater detail below, decoder 262 uses the error correcting code embedded within each trailer group transport stream packet, to correct for errors in TSP₀ through TSP_N. The recovered transport stream 208 is then outputted to MPEG-2 end-user 228.

[0052] Having described system 250, the presently preferred method of forward error correction as used by encoder 254 will now be described with reference to Figures 4-12. Referring now to Figure 4, at step 300 a super group of two-hundred-and-forty-six transport stream packets 30 is defined. In Figure 5, step 300 is shown in greater detail, as two-hundred-and-forty-six transport stream packets (TSP₀ ... TSP₂₄₅) are taken from transport stream 208 and formed into a super group 304. While it has been found that two-hundred-and-forty-six is a presently preferred number of transport stream packets to form super group 304, it is to be understood that other numbers of transport stream packets can be used to form super group 304 without departing from the scope of the invention. Considerations for the selection of different size super groups will be discussed in greater detail below.

[0053] Referring to Figure 4, at step 350 a suitable error correction operation is performed on parallel bytes of each transport stream packet within super group 304. Referring to Figure 6, step 350 is shown in greater detail. Error correction operation 354 is performed on parallel bytes of each transport stream packet (TSP₀, TSP₁, TSP₂, ... TSP₂₄₅) within super group 304, thus generating error correction packets (ECP) (ECP₁, ECP₂, ... ECP₁₈₇) as indicated at 358. For greater clarity, ECP₁ is derived from Byte₁ of TSP₀, TSP₁, TSP₂, through TSP₂₄₅; ECP₂ is derived from Byte₂ of TSP₀, TSP₁, TSP₂, through TSP₂₄₅ etc. The foregoing is presently preferred in order to protect for bursty errors and can thus obviate the need to interleave the transport stream packets within the transport stream. However, it will be understood that error correction operation 354 can be performed on other parallel or non-parallel structures of super group 304, although it is preferred that at least one byte for processing is taken from each TS packet and no byte is processed twice to achieve at least some interleaving of the data.

[0054] Any error correction operation known in the art can be used, but in a presently preferred embodiment, Reed Solomon coding is used to generate error correction packets 358 of nine bytes each. In order to reduce the size of error correction data, it is presently preferred to only perform error correction operation 354 on Byte₀ - Byte₁₈₇, as Byte₀ or Sync Byte 42 within each transport stream packet is always known to be "0x47". (As used herein, "0x...." indicates a number expressed in hexadecimal format.)

[0055] Referring to Figures 4 and 7, at step 400 a trailer group 404 consisting of eleven MPEG-2 compliant transport stream packets is created and added to super group 304. It is to be understood that eleven transport stream packets is particularly suited to accommodate the error correction packets 358 generated at step 350 and accordingly, trailer group 404 can have a fewer or greater number of packets depending on the size of super group 304 and the desired

error correction operation 354.

[0056] As shown in detail in Figure 8, each packet within trailer group 404 (TSP₂₄₆, ... TSP₂₅₆) has an adaptation field 74 fixed at one-hundred-eighty-four bytes in length, and has no payload data 78. Each trailer group 404 packet is made optional by setting transport error indicator 46, payload start indicator 50 and transport priority 54 all to the value of zero. In accordance with the MPEG-2 standard, it is presently preferred that Packet Identifier 58 for each trailer group 404 packet be "0x1FFE" in order to avoid collision of Packet Identifiers 58 in trailer group 404 with other Packet Identifiers 58 in transport stream 208. For greater clarity, such a collision occurs when a trailer group 404 packet has a common Packet Identifier 58 with another packet in super group 304. The identifier is unique to trailer group 404 packets. It is presently preferred that a test be conducted to confirm that no collision will occur, and if a collision is likely to occur, Packet Identifiers 58 in trailer group 404 are changed to the next smaller value, which in the present example is "0x1FFD". If "0x1FFD" is likely to cause a collision, the "0x1FFC" is chosen, and so on. These values are registered with the Program Map Table in accordance with the MPEG-2 standard. Next, transport scrambling control 62 is set to "b00" and adaptation field control 66 is set to "b10" to indicate that the transport stream packet has adaptation field 74 only, and that no payload 78 follows adaptation field 74. By making these packets optional through setting their contents to be private, they will be transparent to MPEG-2 end-user 228 thus causing MPEG-2 end-user 228 to ignore trailer group 404. Having made the trailer group 404 packets optional, continuity counter 70 is set to a value of "b0000" for each transport stream packet in trailer group 404. Adaptation field length 82 is set to a value of one-hundred-and-eighty-three, transport private data flag 110 is set to a value of one, private data length 128 is set to a value of one-hundred-and-eighty-one in preparation for storage of error correction packets 358.

[0057] At step 450 of Figure 4, error correction packets 358 are inserted into the private data fields 132 of trailer group 404 transport stream packets (TSP₂₄₆, TSP₂₄₇, ... TSP₂₅₅). The insertion of error correction packets 358 into TSP₂₄₆ through TSP₂₅₅ is shown in greater detail with reference to Figures 9 and 10. Because a whole number of nine-byte error correction packets 358 do not fit within each one-hundred-and-eighty-one-byte private data field 132 of each trailer group transport stream packet, it is necessary to split some error correction packets 358 between consecutive trailer group packets. For greater clarity, ECP₁ through ECP₂₀ and the first byte of ECP₂₁ are inserted into TSP₂₄₆. The remaining eight bytes of ECP₂₁, the entirety of ECP₂₂ through ECP₄₀ and the first two bytes of ECP₄₁ are inserted into TSP₂₄₇. This process continues until all one-hundred-and-eighty-one bytes of private data 132 in TSP₂₄₆ through TSP₂₅₄ are filled with error correction packets 358. Figure 10 shows the insertion of the remaining error correction packets 358 into the first fifty-four bytes of private data 132 of TSP₂₅₅.

[0058] Referring to Figure 4, at step 500 identifiers or indices 612 are generated to identify the positions of each packet within super group 304. Step 500 is shown in greater detail with reference to Figures 11-11b. Referring first to Figure 11, index information 602 comprises the number 608 of unique packet identifiers in super group 304, a table 616 of unique identifiers and indicators 614, and start continuity counters 620. First, each packet in super group 304 is scanned to determine the number "P" of unique Packet Identifiers 58 in super group 304. "P" is then stored in private data field 132 of TSP₂₅₅, as indicated at 608. Next, indicators 614 (Indicator₁-Indicator_P) are determined, each of which indicates the number of bits that will be required to represent a sequence index 612 for each unique Packet Identifier 58 found in super group 304. It is presently preferred that each indicator 614 is three bits in length and its value is set to four or "b100", thereby indicating that the sequence index 612 for each Packet Identifier 58 is four bits. Table 616 of indicators 614 and their corresponding Packet Identifier 58 (PID₁...PID_P) are stored in private data field 132 of TSP₂₅₅. For greater clarity PID₁ is the first unique Packet Identifier 58 in super group 304 and its corresponding index 612 will be "b0000"; PID₂ is the second unique Packet Identifier 58 in super group 304 (TSP₀, TSP₁,...TSP₂₄₅) and its corresponding index 612 will be "b0001", etc. Because each Packet Identifier is thirteen bits, and each indicator 614 is three bits, the total space required for each pair 616 is sixteen bits or two bytes. Accordingly, in the presently preferred embodiment, a total of two-times-P (2 x P) bytes are required to store all of table 616.

[0059] Next, each of the continuity counters 70 of the first transport stream packet within super group 304 having a unique Packet Identifier 58 are copied sequentially into TSP₂₅₅ to form start continuity counters 620. For greater clarity CC_{PID1} is the first continuity counter 70 of the first transport stream packet having the first unique Packet Identifier 58 (PID₁) in super group 304; CC_{PID2} is the first continuity counter 70 of the first transport stream packet having the second unique Packet Identifier 58 (PID₂) in super group 304, and so on. Start continuity counters 620 requires a total of W bytes, which is determined using Equation 1 in Appendix A.

[0060] Referring now to Figure 11a, sequence indices 612 corresponding to each packet identifier 58 in each sequential transport stream packet of super group 304 from TSP₀ through TSP_Z are stored in TSP₂₅₅ until TSP₂₅₅ is full. Sequence indices 612 require a total of X bytes, which is determined using Equation 2 in Appendix A. Z, the number of transport stream packets whose sequence index is stored in TSP₂₅₅, is determined using Equation 3 in Appendix A. It will be apparent to persons of skill in the art that Z can equal a value of two-hundred-and-forty-five, for example, in the case where all of the transport stream packets in super group 304 have the same Packet Identifier 58, and accordingly all sequence indices 612 would be stored within TSP₂₅₅.

[0061] Referring now to Figure 11b, the remaining sequence indices 612, if any, are stored in TSP₂₅₆. As shown in

Figure 11b, sequence indices 612 require a total of Y bytes, which is determined using Equation 4 in Appendix A. Where Y is less than one-hundred-and-eighty-one bytes, stuffing bytes 156 will be added to the end of TSP₂₅₆ to fill the packet. It will be apparent to persons of skill in the art that where Z has a value of two-hundred-and-forty-five, Y will have a value of zero, and, that the formation of TSP₂₅₆ is optional.

[0062] It will be appreciated by persons skilled in the art that other methods of identifying the location of packets within super group 304 can be performed, such as embedding a unique identifier or absolute sequence number in the private data fields 132 of each of the transport stream packets within super group 304. However, this method can have disadvantages in that it necessitates the creation of new transport stream packets that accommodate the displaced payload data during the process of embedding absolute sequence numbers, although the result is that the error correction information is still optional and therefore transparent to an asymmetric decoder. Another method of identifying the location of packets employs a combination of: embedding absolute sequence numbers in the first packet of sub-groups or intervals of transport stream packets within the super group, and storing sequence identifiers in the trailer group of the remaining packets in each sub-group, relative to the absolute sequence numbers. In this method, it is presently preferred to provide two-hundred-and-forty-six packets in each super group and six packets in each sub-group, and performing a Reed-Solomon operation which, when combined with the generated sequence identifiers, generates a total often packets in the trailer group. This method suffers the same disadvantage as storing absolute sequence numbers in each transport stream packet, as it necessitates the creation of new transport stream packets to accommodate displaced payload data. Other methods of identifying the location of packets within super group 304 will occur to persons of skill in the art, and are within the scope of the present invention.

[0063] It will be understood by persons of skill in the art that the inherent features of TSP₂₅₆ can be used to signify the end of super group 304. For example, simply using packet identifier 58 to identify that TSP₂₄₆ through TSP₂₅₆ are trailer group 404 transport stream packets. However, an "end trailer group" identifier, such as "0x1F1FFE", can be added to the end of the last packet (TSP₂₅₆) of trailer group 404 to indicate the end of trailer group 404 to a symmetric decoder 262. However, the inclusion of an "end trailer group" identifier is not a critical feature of the present invention.

[0064] Steps 300 to 500 are then repeated for subsequent super groups 304 of two-hundred-and-forty-six transport stream packets, thereby generating a continuous enhanced transport stream 258. It is presently preferred to generate and send a pair of super groups 304 and then a pair of corresponding trailer groups 404 as shown in Figure 12 in order to protect for some types of bursty errors, however, this can be eliminated by inserting extra, optional null transport stream packets 412 between super group 304 and the trailer group 404, each containing only stuffing bytes 156, as shown in Figure 12a. It is presently preferred that the number of null packets 412 be equivalent to the maximum number of transport packets lost due to a bursty error.

[0065] It will now be apparent that enhanced transport stream 258 has optional forward error correction information included therein, such that super group 304 can be simply transmitted directly to MPEG-2 end-user 228. Accordingly, super group 304 (TSP₀ ... TSP₂₄₅) can be processed by MPEG-2 end-user 228, and trailer group 354 (TSP₂₄₆...TSP₂₅₆) will be transparent to MPEG-2 end-user 228. Alternately, super group 304 can be transmitted to a symmetric decoder 262 which can decode TSP₀ through TSP₂₄₅ and utilize the error correction data contained within trailer group 404 (TSP₂₄₆...TSP₂₅₆) to reassemble lost packets. As will be apparent, one of the advantages of the present invention is that a broadcaster of an enhanced transport stream 258 does not need to know if the receiver has a suitable decoder.

[0066] It will be apparent from the foregoing that super group 304 can have a fewer or a greater number of transport stream packets than the presently preferred embodiment. There are three main factors that can affect the selection of the size of the super group 304. First, a super group 304 of two-hundred-and-forty-six transport stream packets allows the recovery of up to nine lost transport stream packets or up to four-and-a-half errored transport stream packets. This meets the requirements of an environment requiring a bit error ratio of 10^{-3} to 10^{-2} which is acceptable for both cable and satellite transmissions. A shorter super group 304 with the same length of trailer group 404 would be able to work in an environment having a higher bit error ratio, while a longer super group in an environment of a lower bit error ratio. Second, a super group 304 having two-hundred-and-forty-six transport stream packets has a low overhead of 4.47%. However, a super group 304 with fewer transport stream packets would have a higher percentage of overhead. Finally, a super group with more transport stream packets would result in increased computational complexity for the error correction operation.

[0067] Having described an encoding method in accordance with the preferred embodiment of the present invention, a decoding method symmetric to the encoding method of Figure 4, and in accordance with another embodiment of the present invention will now be described with reference to Figure 13. At Step 700 decoder 262 takes a super group 304 from enhanced transport stream 258, and defines an error group by examining super group 304 to determine the consecutive positions of any lost transport stream packets within super group 304. In the event that more than one consecutive group is located, an error group is defined for each and processed in turn. Table 616, start continuity counters 620 and indices 612 are first accessed from the trailer group 404 received from enhanced transport stream 258. Using these data, Packet Identifiers 58 and continuity counters 70 for each transport stream packet 30 in super group 304 are reconstructed and are sorted into their original order. Next, Packet Identifiers 58 and continuity counters

70 from super group 304 received from enhanced transport stream 258 are compared to the reconstructed values to reveal the positions of lost transport stream packets in super group 304. It will be apparent that, where no transport stream packets are lost, a one-to-one correspondence will be found during the comparison and the process can move directly to step 720. However, if the Packet Identifiers 58 and continuity counters 70 do not match with their corresponding index 612, but instead match the values of some later index 612 in the stored sequence, then the error group is defined as comprising the range of lost consecutive transport stream packets between the first mismatch and the location of where the next match is found. Other error groups can be found within super group 304 by repeating this process.

[0068] At step 720, an error recovery operation is performed to recover all lost transport stream packets within the error group. The error recovery operation is symmetric to error correction operation 354, and in the presently preferred embodiment Reed Solomon decoding is used. The error recovery operation is performed by using corresponding error correction packets 358 embedded within trailer group 404. The individual bit errors within transport stream packet are recovered simultaneously. As the exact positions of bit errors are not directly detectable, such errors are processed by the inherent features of the Reed Solomon bit error decoding operation. Once super group 304 is completely corrected, decoder 262 then outputs super group 304 as a portion of transport stream 208. Steps 700 and 720 are then repeated for subsequent super groups, thereby continuously error correcting enhanced transport stream 258 and outputting transport stream 208.

[0069] In another embodiment of the present invention, error correcting data can be generated and inserted into a private data field 132 of each transport stream packet. Since private data field 132 is optional, an asymmetric decoder will still ignore the data contained within the private data field, but will process the payload data according to the MPEG-2 standard. In this embodiment, part of the payload of each transport stream packet is shifted into new transport stream packets, and then error correction data is embedded within the remaining space of the private data field of each transport stream packet. In other words, each transport stream packet carries some payload and some error correction data. This embodiment is presently less preferred as it requires interleaving of error correction information with the payload in each of the transport stream packets to deal with "bursty" errors and it also requires the generation of additional overhead to accommodate the adaptation field in each transport stream packet.

[0070] In another embodiment of the present invention, error correction data can be embedded within other private data packets, private sections or descriptors etc. as defined within the MPEG-2 standard. For example, the error correction data can be inserted into payload 78 by defining a private packet identifier 58 in the Program Map Table assigning it a value between "0x80" and "0xFF", as more particularly defined in the MPEG-2 standard.

[0071] It will be apparent that the present invention can be implemented by hardware or software, and that it need not be implemented using an independent encoder 254 and decoder 262. Instead, encoder 254 can be integral with an MPEG-2 encoder, and decoder 262 can be integral with the MPEG-2 decoder.

[0072] While only specific combinations of the various features and components of the present invention have been discussed herein, it will be apparent to those of skill in the art that desired sub-sets of the disclosed features and components and/or alternative combinations of these features and components can be utilized, as desired. For example, while the embodiments discussed herein are directed to error correcting data streams such as the transport stream according to the MPEG-2 standard, it is to be understood that the present invention is applicable to other data streams such as the MPEG-1 standard, future MPEG standards or other time-ordered data streams.

[0073] The present invention provides a novel system and method for a novel forward error correction method and system to protect a digital transport stream, by providing an MPEG-2 compliant enhanced transport stream with optional forward error correction data included therein. Accordingly, the present invention can be used as a means of forward error correcting an MPEG2 transport stream for broadcast in any of the Digital Video Broadcast formats without the need for effectively creating another protocol layer in order to add error correction. This results in an error correction system and method that does not require decoders symmetric to the error correction encoders, and yet provides the flexibility of providing an error correcting decoder to improve the reliability of an MPEG-2 transport stream.

[0074] It is believed that the present invention results in a system and method for forward error correction with less delay than current systems and methods known in the art, as the transport stream packets need not be interleaved. It is further believed that strong error recovery is available as up to nine lost transport stream packets and up to four-and-a-half errored packets can be corrected in a presently preferred embodiment. It is also believed that the present invention results in a forward error correcting system and method with low overhead, as only 4.47% of data is added, while still achieving powerful error recovery ability.

Appendix A

5

Equation 1

10

$$W = \text{int} (P/2 + 0.5)$$

15

Equation 2

$$X = \min (246 \times 4 / 8, 126 - 2 \times P - W) = \min (123, 126 - 2 \times P - W)$$

20

Equation 3

25

$$Z = 2 \times X - 1$$

30

Equation 4

$$Y = \max (0, 246 \times 4 / 8 - (126 - 2 \times P - W)) = \max (0, 2 \times P + W - 3)$$

35

Claims

40

1. A method of creating a portion of an enhanced MPEG compatible data stream with optional forward error correction information included therein, comprising the steps of:

45

(i) receiving an MPEG data stream of data packets from an MPEG data stream source;

(ii) forming a super group including a predefined number of said data packets;

(iii) performing a selected forward error correcting operation on said super group of packets to obtain error correction data;

50

(iv) creating index data which relates said error correction data to one or more corresponding super group data packets;

(v) creating a trailer group including a sufficient number of MPEG compliant packets to accommodate said error correction data and said index data, said trailer group further including code to identify each packet as optional;

55

(vi) arranging said error correction data and said index data into said trailer group; and

(vii) assembling said MPEG compatible data stream with at least one super group followed by at least one

trailer group corresponding with said at least one super group.

2. A method of transmitting data as an enhanced MPEG compatible data stream with optional forward error correction information therein, comprising the steps of:

- (i) receiving an MPEG data stream of data packets from an MPEG data stream source;
- (ii) forming a super group including a predefined number of said data packets;
- (iii) performing a selected forward error correcting operation on said super group of packets to obtain error correction data;
- (iv) creating index data which relates said error correction data to one or more corresponding super group data packets;
- (v) creating a trailer group including a sufficient number of MPEG compliant packets to accommodate said error correction data and said index data, said trailer group further including code to identify each packet as optional;
- (vi) arranging said error correction data and said index data into said trailer group;
- (vii) assembling said enhanced MPEG compatible data stream with at least one super group followed by at least one trailer group corresponding with said at least one super group;
- (viii) broadcasting said enhanced MPEG compatible data stream;
- (ix) receiving said enhanced MPEG compatible data stream;
- (x) extracting at least one received super group and at least one received trailer group corresponding to said at least one received super group from said data stream received in step (ix); and
- (xi) performing an error recovery operation on said at least one received super group using said error correction data in said at least one corresponding received trailer group.

3. The method according to claim 1 or 2, wherein said assembly in step (vii) comprises two super groups following by two corresponding trailer groups.

4. The method according to claim 1 or 2, wherein said assembly step (vii) comprises said super group followed by a predefined number of null packets and said trailer group.

5. The method according to claim 4, wherein said predefined number is equivalent to an estimated maximum burst error of the broadcast environment.

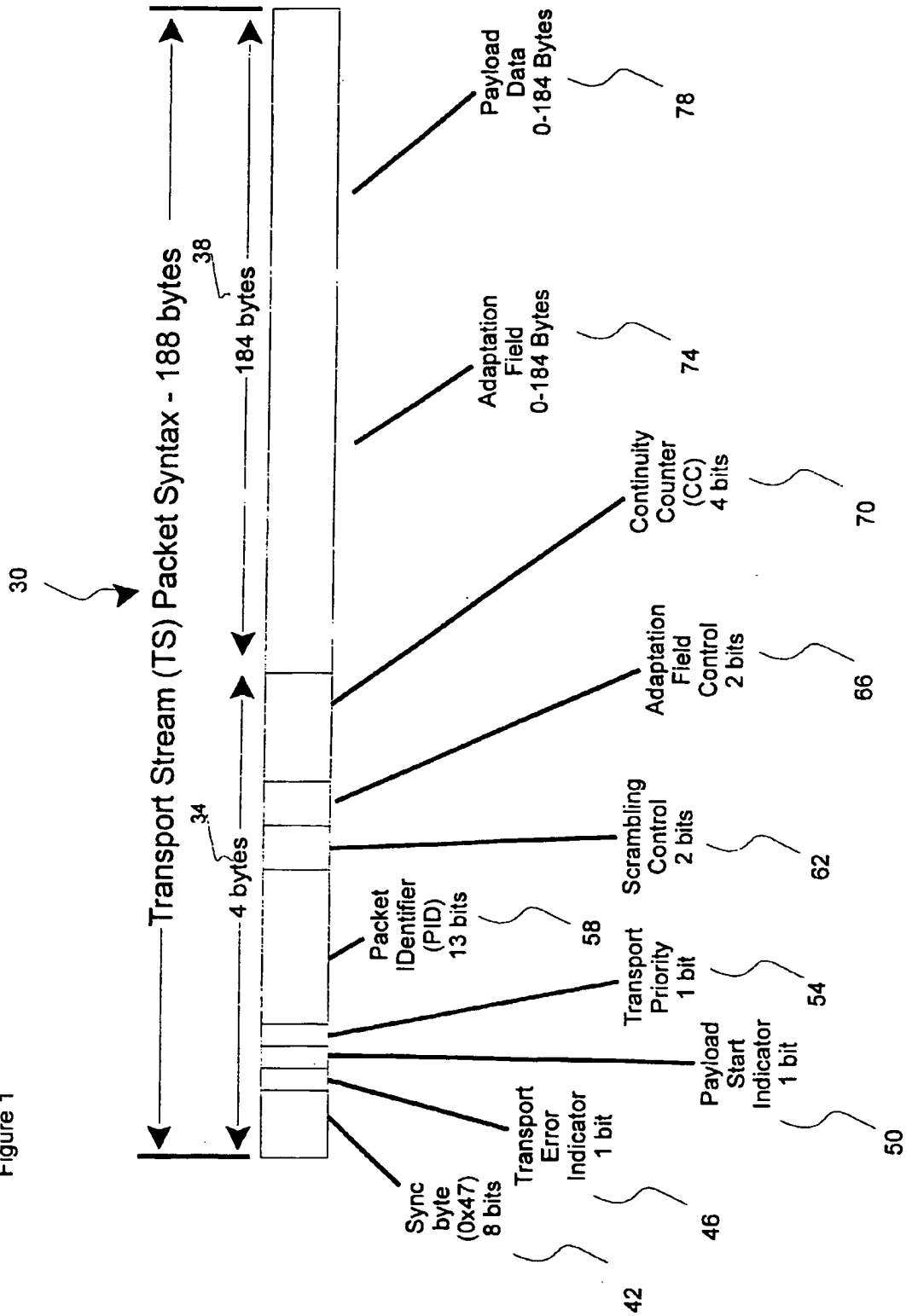
6. The method according to any preceding claim, wherein said error correction operation is performed on parallel bytes of each packet within said super group.

7. A method of recovering a portion of an enhanced MPEG compatible data stream of data packets having super groups of data packets and corresponding trailer groups of optional packets having forward error correction information and indexing information included therein, comprising the steps of:

- (i) Receiving said enhanced MPEG compatible data stream;
- (ii) extracting at least one received super group and at least one received trailer group corresponding to said at least one received super group from said data stream received in step (i);
- (iii) performing an error recovery operation on said at least one received super group using said error correction data and said indexing information in said at least one corresponding received trailer group.

8. The method according to any preceding claim, wherein said data stream is an MPEG-2 transport stream.
9. The method according to any preceding claim, wherein said super group comprises two-hundred-and-forty-six data packets.
10. The method according to any preceding claim, wherein said trailer group comprises eleven data packets.
11. The method according to any preceding claim, wherein said error correction or error recovery operation is Reed Solomon coding.
12. The method according to any preceding claim, wherein each of said packets in said super group includes a packet identifier and each of said packets in said trailer group includes a packet identifier different from said super group packet identifiers.
13. A portion of an enhanced MPEG compatible data stream with optional forward error correction information included therein, comprising:
- (i) at least one super group including a predefined number of data packets;
 - (ii) at least one trailer group including a predefined number of MPEG compliant packets, said at least one trailer group further including code to identify each packet as optional;
 - (iii) error correction data embedded within said at least one trailer group, said error correction data being obtained from a selected forward error correcting operation performed on at least one corresponding super group of data packets; and
 - (iv) index data embedded within said at least one trailer group which relates said error correction data to one or more said corresponding super group data packets.
14. The data stream according to claim 13, wherein said data stream is an MPEG-2 transport stream.
15. The data stream according to claim 13 or 14, wherein said super group comprises two-hundred-and-forty-six data packets.
16. The data stream according to any one of claims 13 to 15, wherein said trailer group comprises eleven data packets.
17. The data stream according to any one of claims 13 to 16, wherein said forward error correction operation is Reed Solomon coding.
18. The data stream according to any one of claims 13 to 17, wherein said error correction data was generated from parallel bytes of each packet within said super group.
19. The data stream according to any one of claims 13 to 18, wherein each of said packets in said super group includes a packet identifier and each of said packets in said trailer group includes a packet identifier different from said super group packet identifiers.

Figure 1



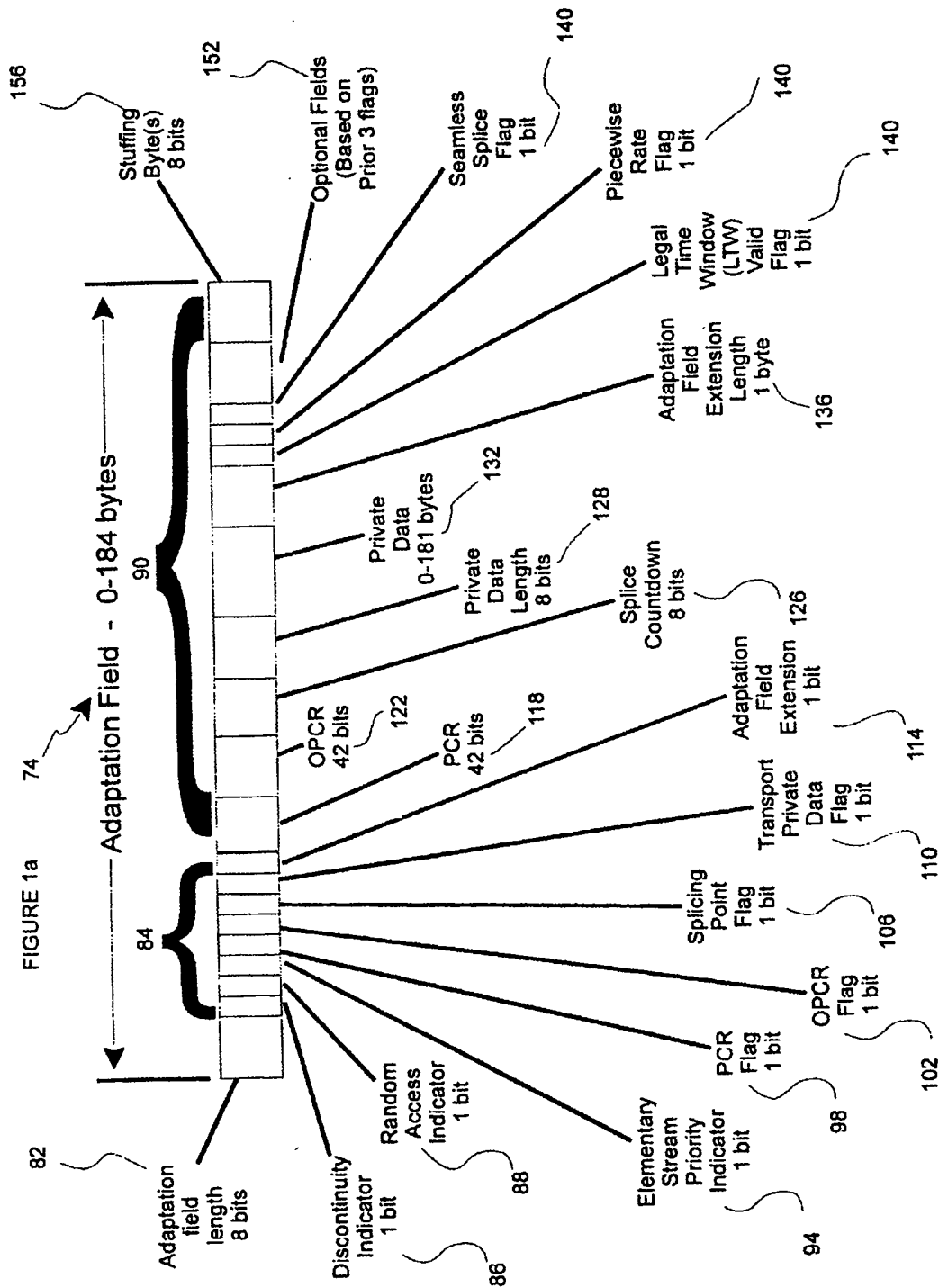


Figure 1b

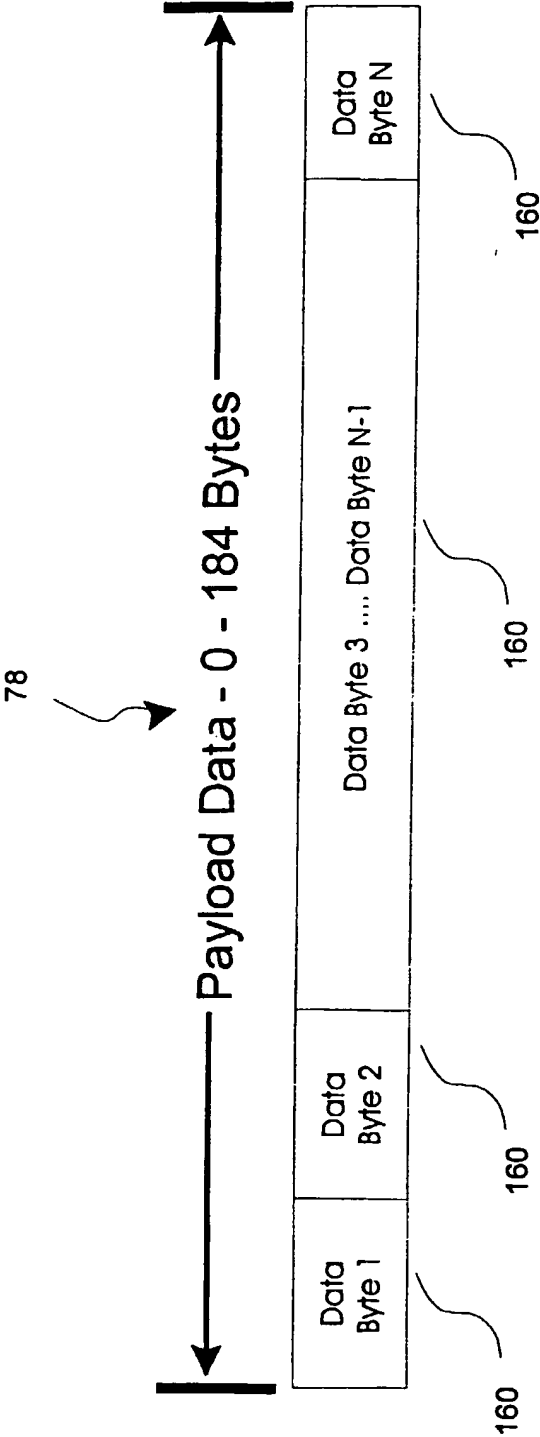


Figure 2 (Prior Art)

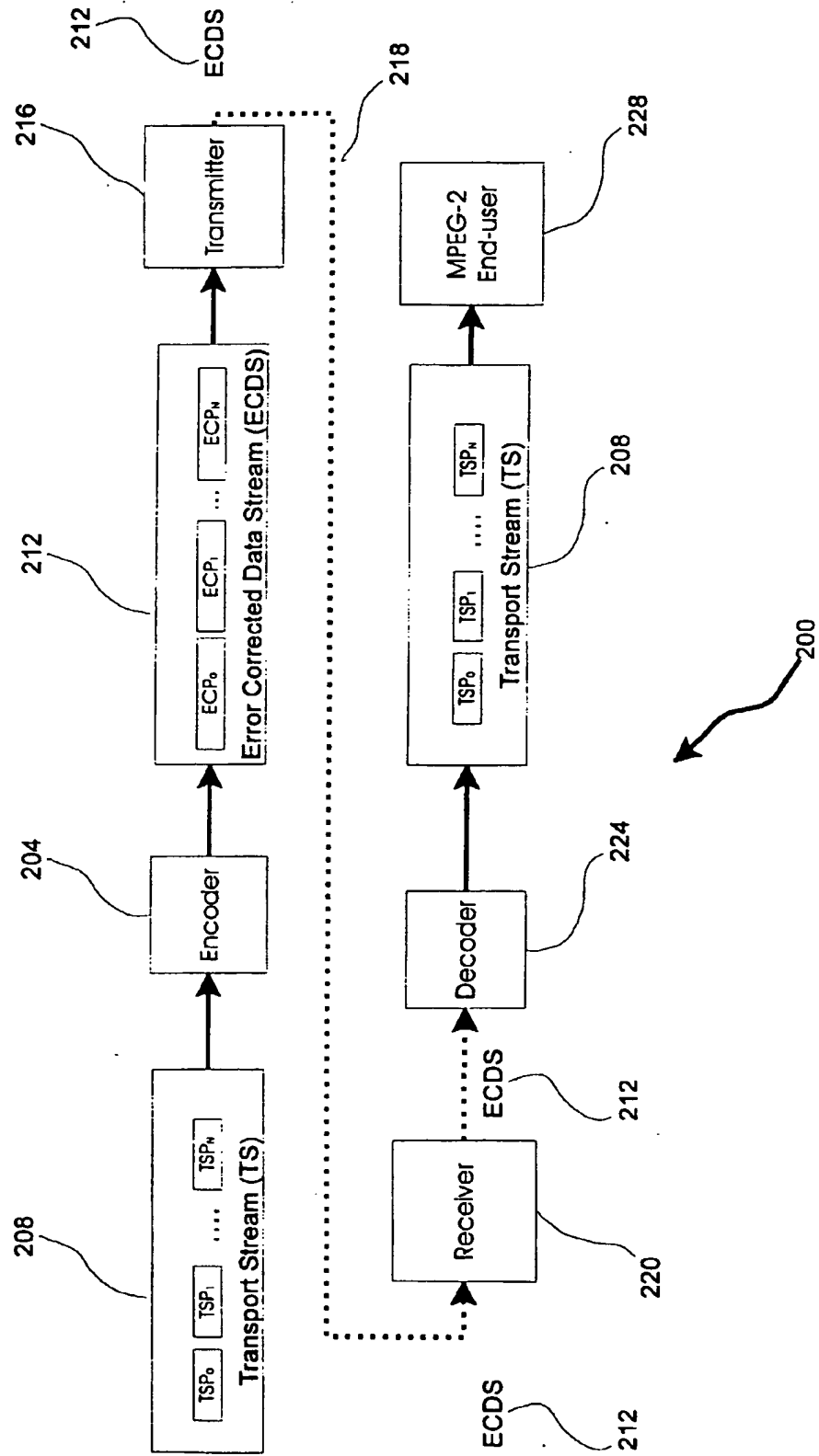


Figure 3

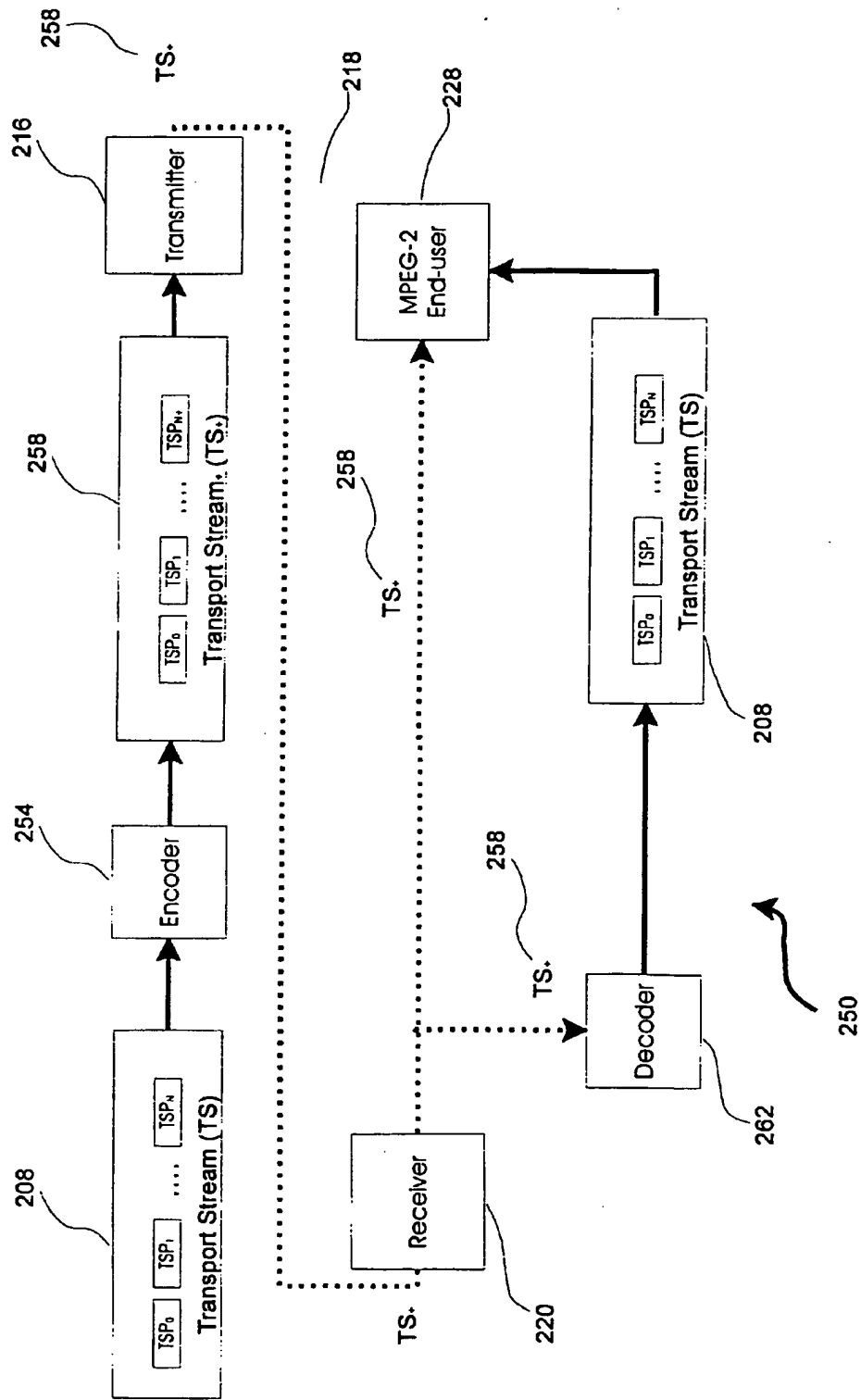


Figure 4

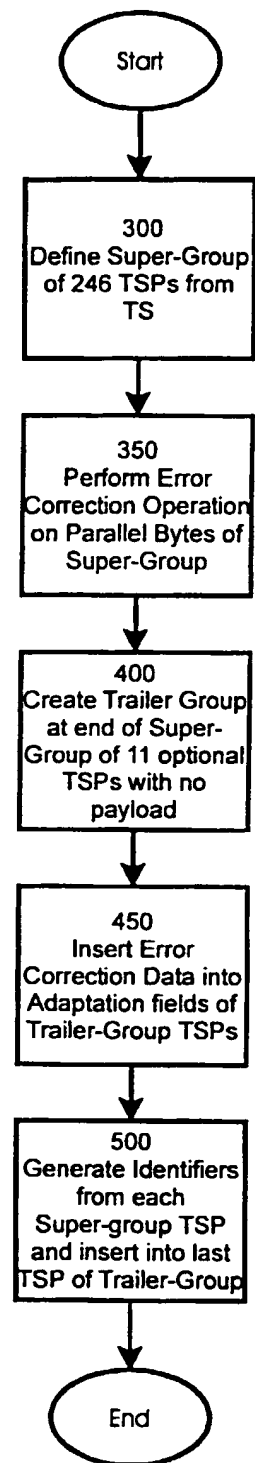


Figure 5
(Step 300)

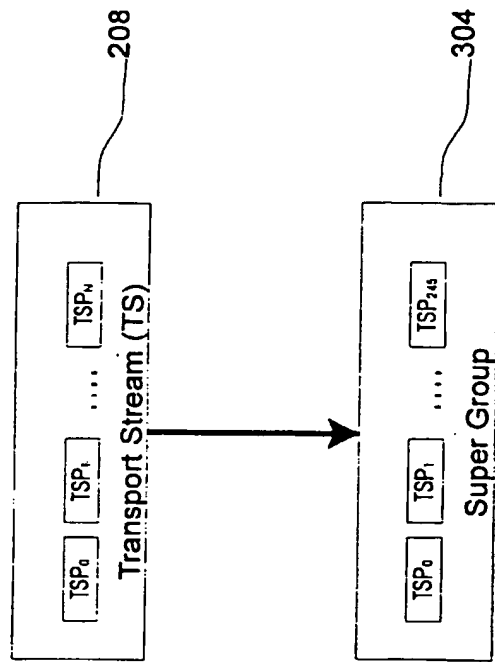


Figure 6
(Step 350)

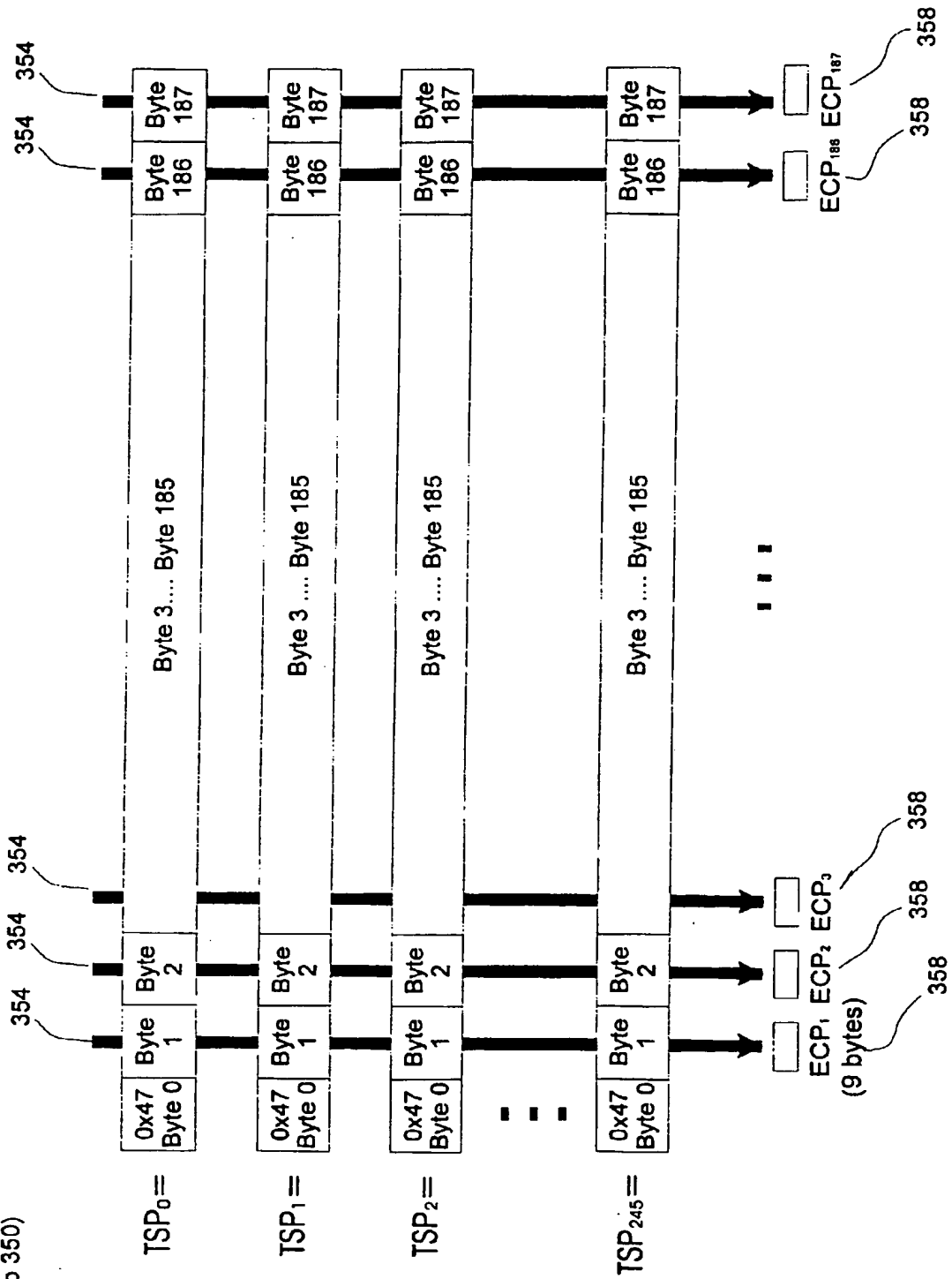


Figure 7
(Step 400)

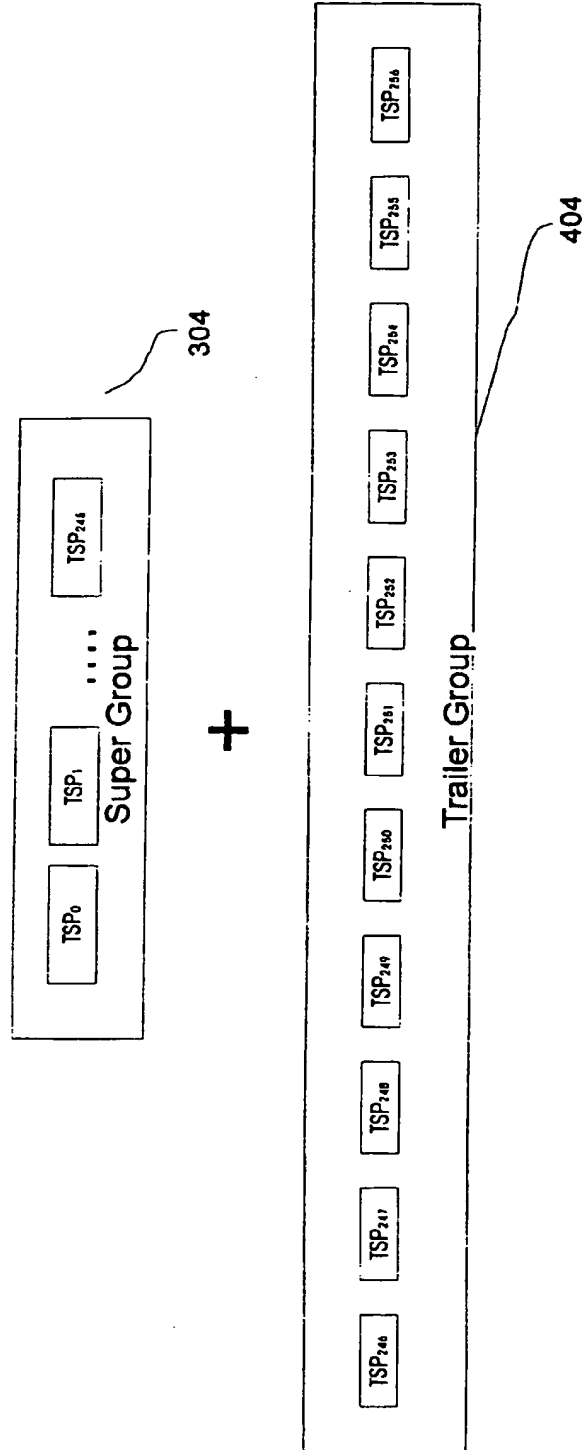
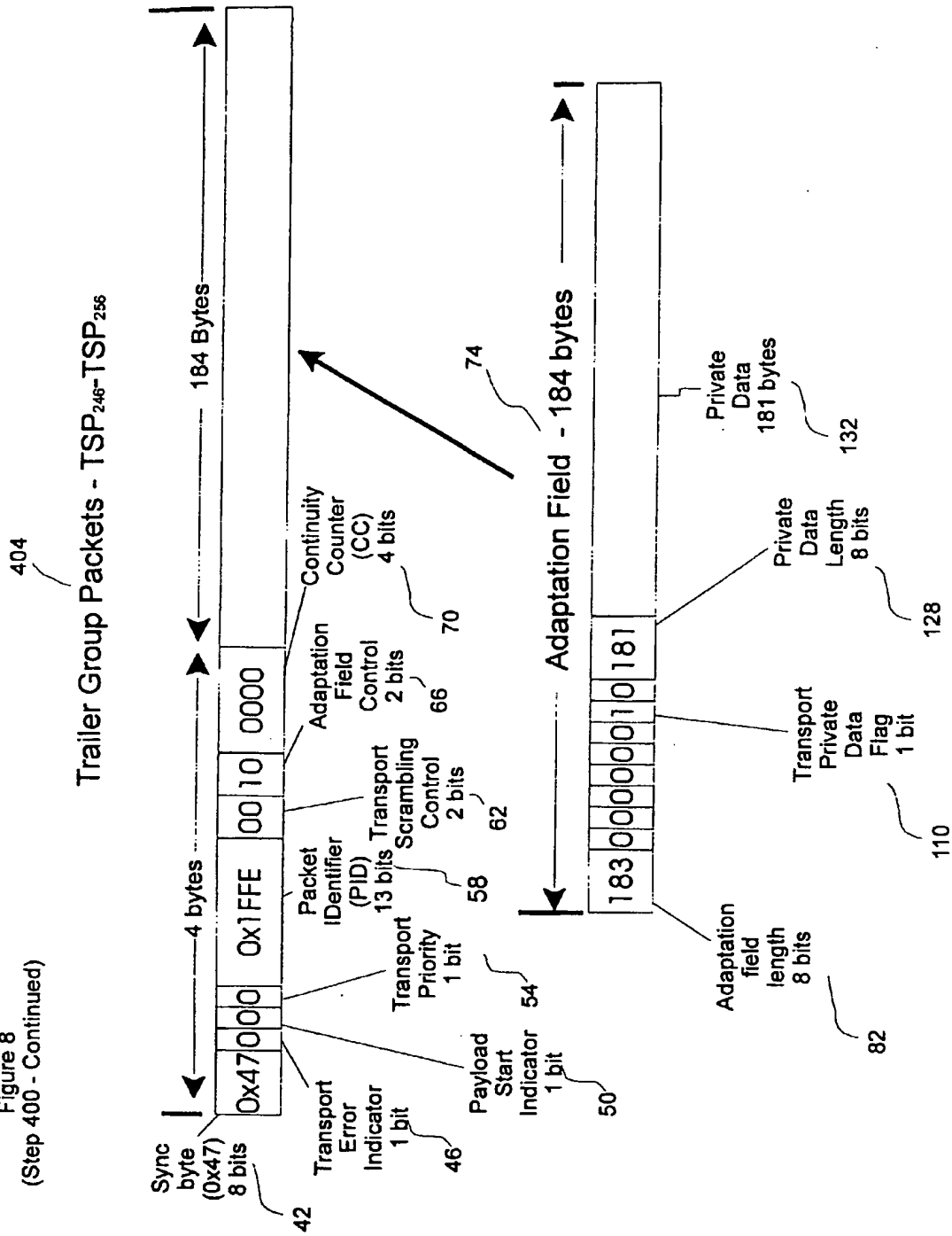
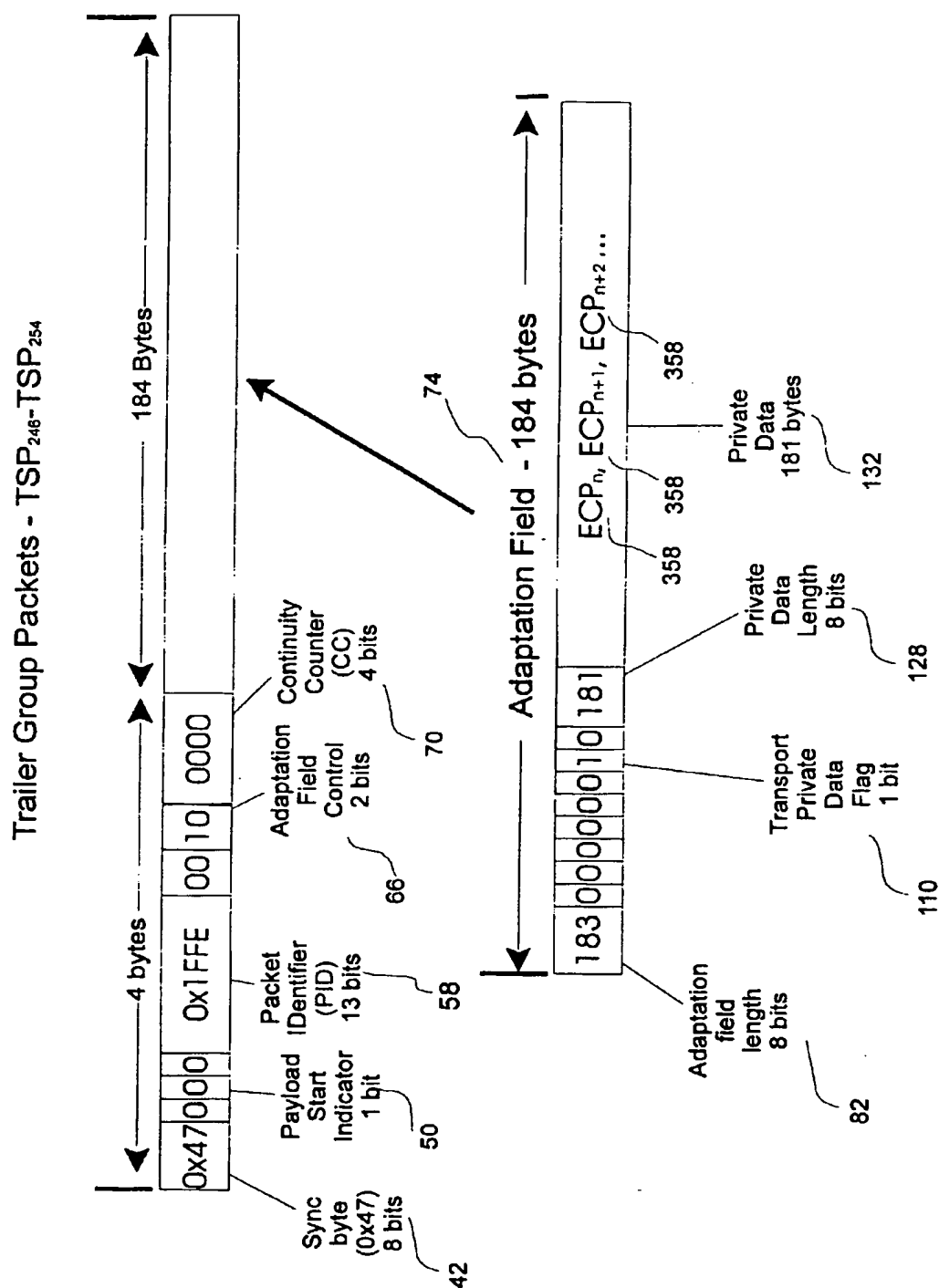


Figure 8
(Step 400 - Continued)



**Figure 9
(Step 450)**



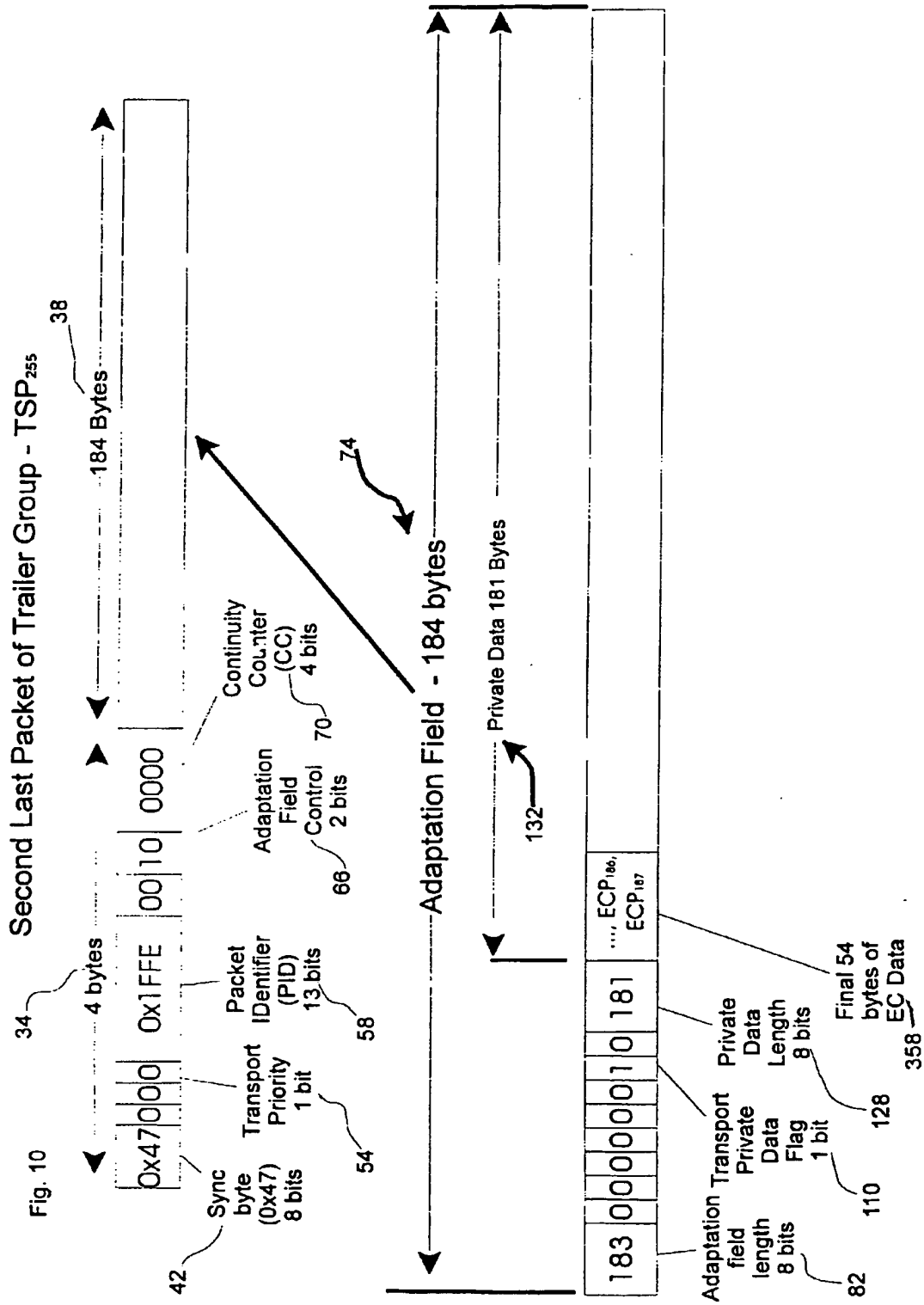
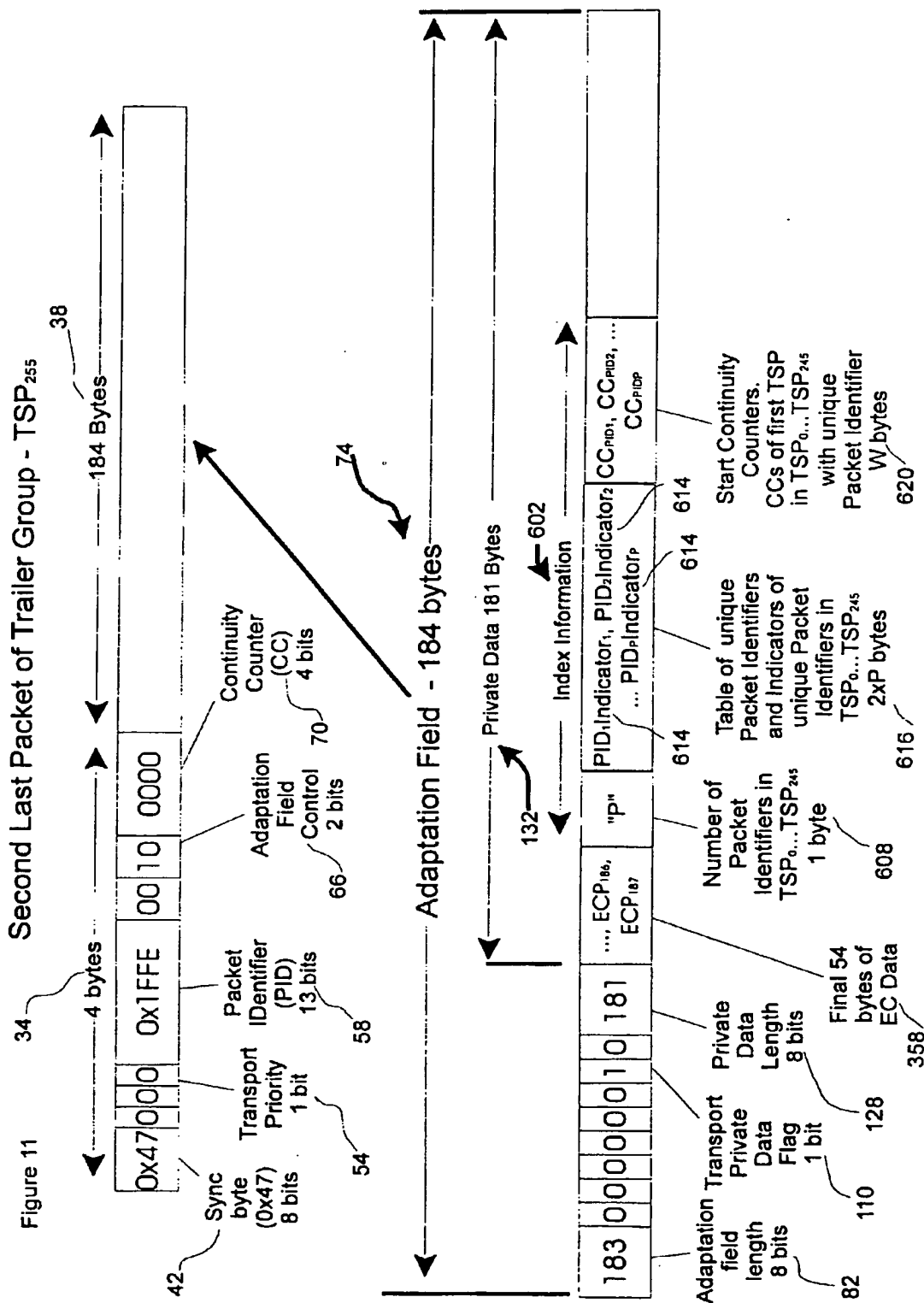
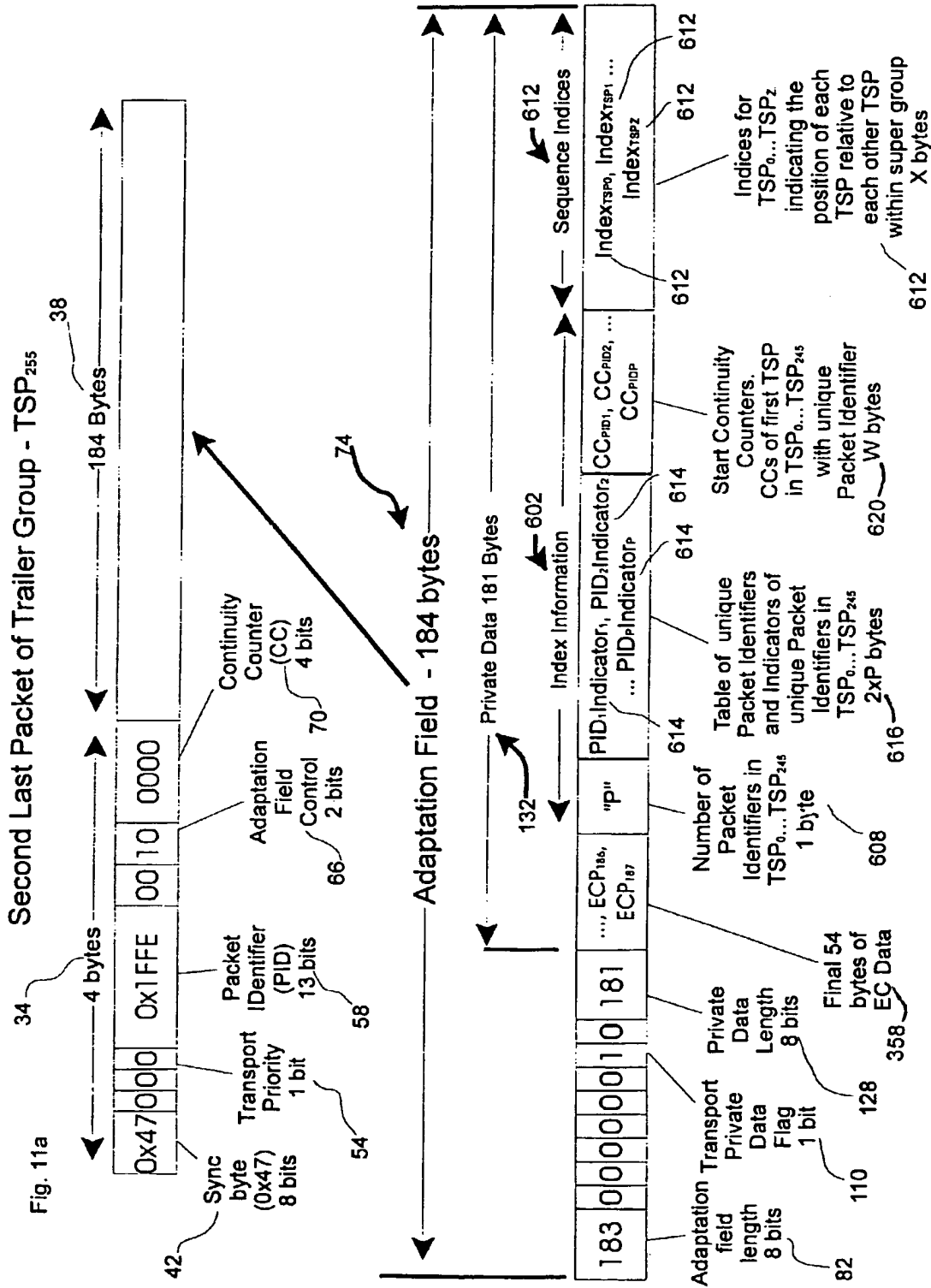
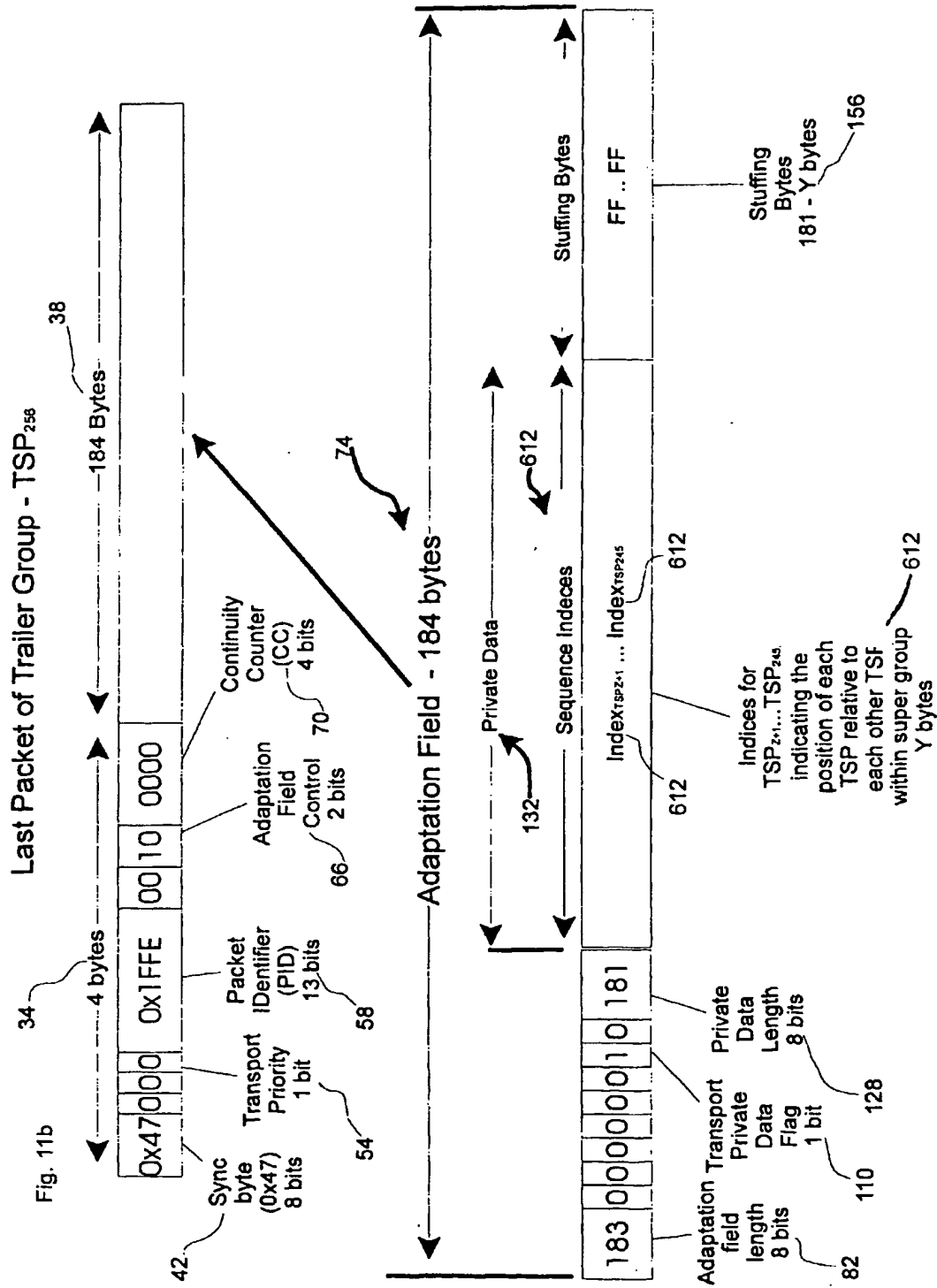
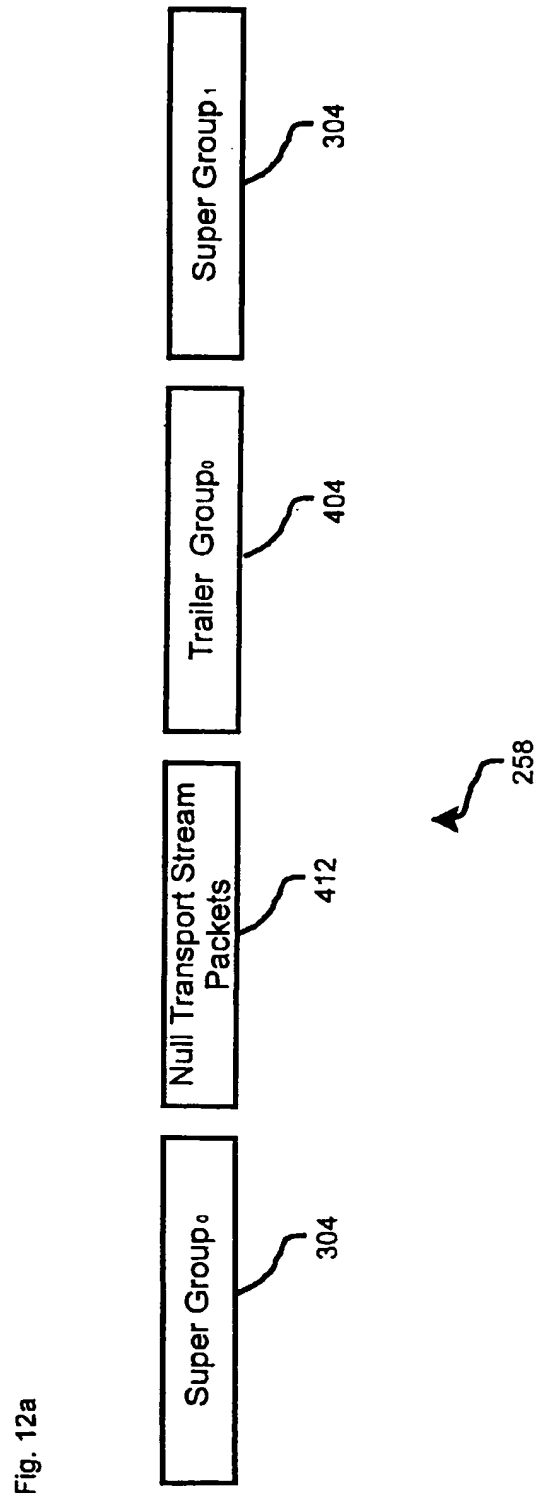
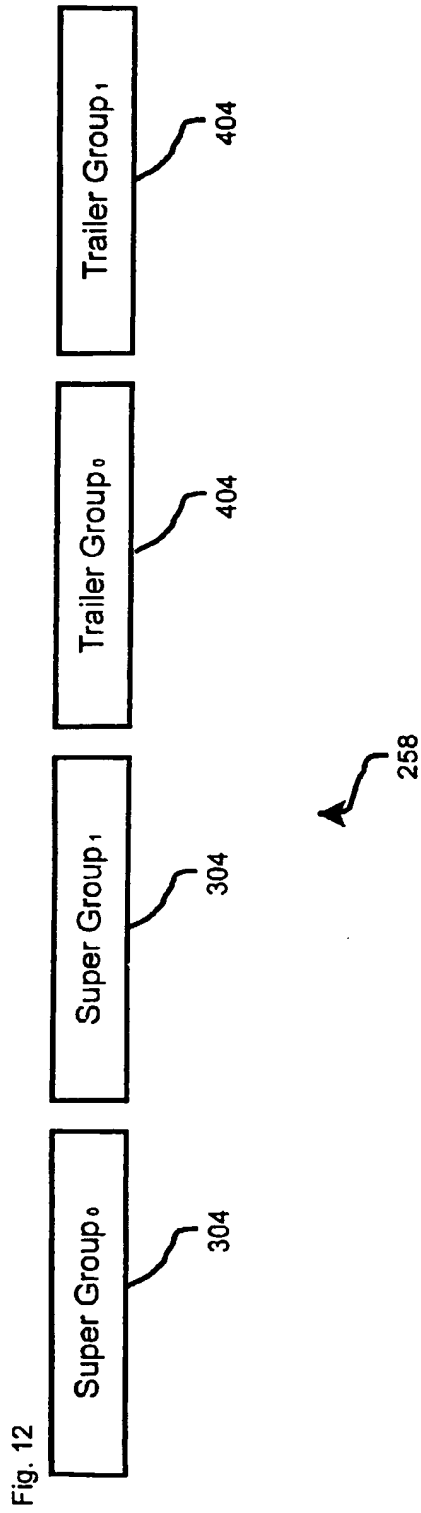


Figure 11









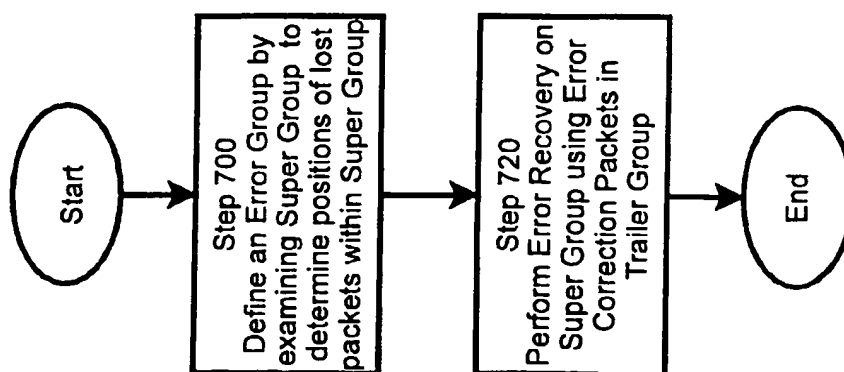


Fig. 13



European Patent
Office

EUROPEAN SEARCH REPORT

Application Number
EP 99 31 0433

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
Y	WO 98 52323 A (HIGASHIDA MASAOKI; KOBAYASHI MASAOKI (JP); MORIOKA YOSHIHIRO (JP);) 19 November 1998 (1998-11-19) * abstract * * figures 1-3,5,15-17,39,40 *	1,2,6-8, 11-14, 17-19	H04N7/66 H04N7/24 H04N7/52
P, Y	& EP 0 928 086 A (MATSUSHITA ELECTRIC IND CO LTD) 7 July 1999 (1999-07-07) * abstract * * column 12, line 50 - column 15, line 40 * * column 25, line 10 - column 27, line 2 * * column 52, line 29 - column 54, line 54 * * figures 1-3,5,15-17,39,40 *	1,2,6-8, 11-14, 17-19	
Y	EP 0 880 277 A (SONY UK LTD) 25 November 1998 (1998-11-25) * the whole document *	1,2,6-8, 11-14, 17-19	
A	DRURY G M: "DVB CHANNEL CODING STANDARDS FOR BROADCASTING COMPRESSED VIDEO SERVICES" ELECTRONICS AND COMMUNICATION ENGINEERING JOURNAL, 6B, INSTITUTION OF ELECTRICAL ENGINEERS, LONDON, vol. 9, no. 1, 1 February 1997 (1997-02-01), pages 11-20, XP000722904 ISSN: 0954-0695 * the whole document *	1-19	TECHNICAL FIELDS SEARCHED (Int.Cl.7) H04N
-/--			
The present search report has been drawn up for all claims			
Place of search THE HAGUE		Date of completion of the search 18 April 2000	Examiner Barel-Faucheux, C
CATEGORY OF CITED DOCUMENTS		T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document	
X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document			

EPO FORM 1503 03.82 (P04031)



European Patent
Office

EUROPEAN SEARCH REPORT

Application Number
EP 99 31 0433

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
D,A	Retrieved from Internet the 14/04/2000 from URL : "http://www.tmo.hp.com/tmo/pia/component_test/PIAApp/Notes/English/MPEGpaper2.html" Protecting MPEG-2 : FEC Schemes in DVB-C,S,T,("Protecting MPEG-2") Copyright 1944-1998 Hewlett-Packard Company XP002135979 * the whole document *	1-19	
			TECHNICAL FIELDS SEARCHED (Int.Cl.7)
The present search report has been drawn up for all claims			
Place of search THE HAGUE		Date of completion of the search 18 April 2000	Examiner Barel-Faucheux, C
<p>CATEGORY OF CITED DOCUMENTS</p> <p>X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document</p> <p>T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document</p>			

EPO FORM 1503 03.82 (P04001)

**ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.**

EP 99 31 0433

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

18-04-2000

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9852323 A	19-11-1998	CN 1228221 T EP 0928086 A	08-09-1999 07-07-1999
EP 0880277 A	25-11-1998	GB 2325595 A JP 11032060 A	25-11-1998 02-02-1999

EPO FORM P0433

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.